# Inference Server

## Release Notes

# Table of Contents

# Chapter 1. Triton Inference Server Overview

Triton Inference Server provides a cloud and edge inferencing solution optimized for both CPUs and GPUs. Triton supports an HTTP/REST and GRPC protocol that allows remote clients to request inferencing for any model being managed by the server. For edge deployments, Triton is available as a shared library with a C API that allows the full functionality of Triton to be included directly in an application.

The Triton Inference Server itself is included in the Triton Inference Server container. External to the container, there are additional C++ and Python client libraries, and additional documentation at GitHub: Inference Server.

This document describes the key features, software enhancements and improvements, any known issues, and how to run this container.

# Chapter 2. Pulling A Container

## About this task

Before you can pull a container from the NGC container registry, you must have Docker installed. For DGX users, this is explained in Preparing to use NVIDIA Containers Getting Started Guide.

For users other than DGX, follow the NVIDIA® GPU Cloud™ (NGC) container registry installation documentation based on your platform.

You must also have access and be logged into the NGC container registry as explained in the NGC Getting Started Guide.

The deep learning frameworks are stored in the following repository where you can find the NGC Docker containers.

**`nvcr.io/nvidia`**

The deep learning framework containers are stored in the `nvcr.io/nvidia` repository.

# Chapter 3. Running The Triton Inference Server

## About this task

To quickly get up-and-running with Triton Inference Server, refer to the Triton Inference Server documentation.

# Chapter 4. Triton Inference Server Release 21.10

The Triton Inference Server container image, release 21.10, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.4.2 with cuBLAS 11.6.5.2
- NVIDIA cuDNN 8.2.4.15
- NVIDIA NCCL 2.11.4 (optimized for NVLink™)
- rdma-core 36.0
- OpenMPI 4.1.1+
- OpenUCX 1.11.1+
- GDRCopy 2.3
- NVIDIA HPC-X 2.9
- TensorRT 8.0.3.4 for x64 Linux
- TensorRT 8.0.2.2 for ARM SBSA Linux
- SHARP 2.5

## Driver Requirements

Release 21.10 is based on NVIDIA CUDA 11.4.2 with cuBLAS 11.6.5.2, which requires NVIDIA Driver release 470 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of

supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.10 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Rate limiter is now available and manages the rate at which requests are scheduled on model instances by Triton.
- Starting with the 21.10 release, a beta version of the Triton Inference Server container is available for the ARM SBSA platform.
- Windows Triton build now supports HTTP protocol.
- Triton added support for caching responses to inference requests.
- Sequence IDs can now accept strings.
- Container composer tool can generate CPU-only Triton containers.
- Refer to the 21.10 column of the Frameworks Support Matrix for container image versions that the 21.10 inference server container is based on.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.10 | 2.15.0 | 20.04 | NVIDIA CUDA 11.4.2 with cuBLAS 11.6.5.2 | TensorRT 8.0.3.4 for x64 Linux<br>TensorRT 8.0.2.2 for ARM SBSA Linux |
| 21.09 | 2.14.0 | | NVIDIA CUDA 11.4.2 | TensorRT 8.0.3 |
| 21.08 | 2.13.0 | | NVIDIA CUDA 11.4.1 | TensorRT 8.0.1.6 |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.07 | 2.12.0 | | NVIDIA CUDA 11.4.0 | |
| 21.06.1 | 2.11.0 | | NVIDIA CUDA 11.3.1 | TensorRT 7.2.3.4 |
| 21.06 | | | | |
| 21.05 | 2.10.0 | | NVIDIA CUDA 11.3.0 | |
| 21.04 | 2.9.0 | | | |
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Traced models in PyTorch seem to create overflows when int8 tensor values are transformed to int32 on the GPU. See https://github.com/pytorch/pytorch/issues/66930.
- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).
- Triton cannot retrieve GPU metrics with MIG-enabled GPU devices (A100 and A30).
- Triton metrics may not work if the host machine is running a separate DCGM agent, either on bare-metal or in a container.
- There is a known issue in TensorRT 8.0 regarding accuracy for a certain case of int8 inferencing on A40 and similar GPUs. The version of TF-TRT in TF2 21.09 includes a feature that works around this issue, but TF1 21.09 does not include that feature and therefore Triton users may experience the accuracy drop for a small subset of model/data type/batch size combinations on A40 when TF-TRT is used through the TF1 backend. This will be fixed in the next version of TensorRT.
- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.

# Chapter 5. Triton Inference Server Release 21.09

The Triton Inference Server container image, release 21.09, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.4.2
- cuBLAS 11.6.1.51
- NVIDIA cuDNN 8.2.4.15
- NVIDIA NCCL 2.11.4 (optimized for NVLink™)
- rdma-core 36.0
- OpenMPI 4.1.1+
- OpenUCX 1.11.1+
- GDRCopy 2.3
- NVIDIA HPC-X 2.9
- TensorRT 8.0.3

## Driver Requirements

Release 21.09 is based on NVIDIA CUDA 11.4.2, which requires NVIDIA Driver release 470 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.09 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Full-featured, beta version of Business Logic Scripting released.
- Beta version for basic JAVA Client released. See https://github.com/triton-inference-server/client/tree/r21.09/src/java for a list of supported features.
- A stack trace is now printed when Triton crashes to aid in debugging.
- The Triton Client SDK wheel file is now available directly from PyPI for both Ubuntu and Windows.
- The TensorRT backend is now an optional part of Triton just like all the other backends. The compose utility can be used to create a Triton container that does not contain the TensorRT backend.
- Model Analyzer can profile with perf_analyzer's C-API.
- Model Analyzer can use the CUDA Device Index in addition to the GPU UUID in the `--gpus` flag.
- Refer to the 21.09 column of the Frameworks Support Matrix for container image versions that the 21.09 inference server container is based on.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.09 | 2.14.0 | 20.04 | NVIDIA CUDA 11.4.2 | TensorRT 8.0.3 |
| 21.08 | 2.13.0 | | NVIDIA CUDA 11.4.1 | TensorRT 8.0.1.6 |
| 21.07 | 2.12.0 | | NVIDIA CUDA 11.4.0 | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.06.1 | 2.11.0 | | NVIDIA CUDA 11.3.1 | TensorRT 7.2.3.4 |
| 21.06 | | | | |
| 21.05 | 2.10.0 | | NVIDIA CUDA 11.3.0 | |
| 21.04 | 2.9.0 | | | |
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event

fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).

- Triton cannot retrieve GPU metrics with MIG-enabled GPU devices (A100 and A30).
- Triton metrics may not work if the host machine is running a separate DCGM agent, either on bare-metal or in a container.
- There is a known issue in TensorRT 8.0 regarding accuracy for a certain case of int8 inferencing on A40 and similar GPUs. The version of TF-TRT in TF2 21.09 includes a feature that works around this issue, but TF1 21.09 does not include that feature and therefore Triton users may experience the accuracy drop for a small subset of model/data type/batch size combinations on A40 when TF-TRT is used through the TF1 backend. This will be fixed in the next version of TensorRT.
- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.

# Chapter 6. Triton Inference Server Release 21.08

The Triton Inference Server container image, release 21.08, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.4.1
- cuBLAS 11.5.4
- NVIDIA cuDNN 8.2.2.6
- NVIDIA NCCL 2.10.3 (optimized for NVLink™)
- rdma-core 36.0
- OpenMPI 4.1.1+
- OpenUCX 1.11.1+
- GDRCopy 2.2
- NVIDIA HPC-X 2.9
- TensorRT 8.0.1.6

## Driver Requirements

Release 21.08 is based on NVIDIA CUDA 11.4.1, which requires NVIDIA Driver release 470 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.08 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Initial Beta release for Business Logic Scripting, a new set of utility functions that allow the execution of inference requests on other models being served by Triton as part of executing a Python model.
- Released new Container Composition Utility which can be used to create custom Triton containers with specific backends and repository agents.
- Starting in 21.08, Triton will release two new containers on NGC.
  - `nvcr.io/nvidia/tritonserver:21.08-tf-python-py3` - GPU enabled Triton server with only the TensorFlow 2.x and Python backends.
  - `nvcr.io/nvidia/tritonserver:21.08-pyt-python-py3` - GPU enabled Triton server with only the PyTorch and Python backends.
- Added Model Analyzer support for models with custom operations.
- Refer to the 21.08 column of the Frameworks Support Matrix for container image versions that the 21.08 inference server container is based on.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.08 | 2.13.0 | 20.04 | NVIDIA CUDA 11.4.1 | TensorRT 8.0.1.6 |
| 21.07 | 2.12.0 | | NVIDIA CUDA 11.4.0 | |
| 21.06.1 | 2.11.0 | | NVIDIA CUDA 11.3.1 | TensorRT 7.2.3.4 |
| 21.06 | | | | |
| 21.05 | 2.10.0 | | NVIDIA CUDA 11.3.0 | |
| 21.04 | 2.9.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option

in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).

- Loading models in ONNX Runtime on the Windows build of Triton may be slow due to the JIT compiler being invoked for newer CUDA architectures. For more information, refer to https://github.com/triton-inference-server/onnxruntime_backend/issues/58/.
- There is a known issue in TensorRT 8.0 regarding accuracy for a certain case of int8 inferencing on A40 and similar GPUs. The version of TF-TRT in TF2 21.08 includes a feature that works around this issue, but TF1 21.08 does not include that feature and therefore Triton users may experience the accuracy drop for a small subset of model/data type/batch size combinations on A40 when TF-TRT is used through the TF1 backend. This will be fixed in the next version of TensorRT.
- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.
- Compared with the 21.02 and earlier releases, there are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The `utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

# Chapter 7. Triton Inference Server Release 21.07

The Triton Inference Server container image, release 21.07, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.4.0
- cuBLAS 11.5.2.43
- NVIDIA cuDNN 8.2.2.6
- NVIDIA NCCL 2.10.3 (optimized for NVLink™)
- rdma-core 32.1
- OpenMPI 4.1.1rc1
- OpenUCX 1.10.1
- GDRCopy 2.2
- NVIDIA HPC-X 2.8.2rc3
- TensorRT 8.0.1.6

## Driver Requirements

Release 21.07 is based on NVIDIA CUDA 11.4.0, which requires NVIDIA Driver release 470 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.07 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Added support for CPU in RAPIDS FIL Backend.
- Inference requests using the C API are now allowed to provide multiple copies of an input tensor in different memories. Triton will choose the most performant copy to use depending on where the inference request is executed.
- For ONNX models using TensorRT acceleration, the `tensorrt_accelerator` option in the model configuration can now specify precision and workspace size.https://github.com/triton-inference-server/server/blob/main/docs/optimization.md#onnx-with-tensorrt-optimization.
- Model Analyzer added an offline mode, which prioritizes throughput over latency for offline inferencing scenarios. A new set of reports and graphs are created to better analyze the offline use case.
- Refer to the 21.07 column of the Frameworks Support Matrix for container image versions that the 21.07 inference server container is based on.
- Ubuntu 20.04 with June 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.07 | 2.12.0 | 20.04 | NVIDIA CUDA 11.4.0 | TensorRT 8.0.1.6 |
| 21.06.1 | 2.11.0 | | NVIDIA CUDA 11.3.1 | TensorRT 7.2.3.4 |
| 21.06 | | | | |
| 21.05 | 2.10.0 | | NVIDIA CUDA 11.3.0 | |
| 21.04 | 2.9.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option

in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).

- The 21.07 release includes libsystemd and libudev versions that have a known vulnerability that was discovered late in our QA process. See CVE-2021-33910 for details. This will be fixed in the next release.
- ONNX Runtime TRT support was removed due to incompatibility with TensorRT 8.0.
- There is a known issue in TensorRT 8.0 regarding accuracy for a certain case of int8 inferencing on A40 and similar GPUs. The version of TF-TRT in TF2 21.07 includes a feature that works around this issue, but TF1 21.07 does not include that feature and therefore Triton users may experience the accuracy drop for a small subset of model/data type/batch size combinations on A40 when TF-TRT is used through the TF1 backend. This will be fixed in the next version of TensorRT.
- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.
- Compared with the 21.02 and earlier releases, there are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The `utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

# Chapter 8. Triton Inference Server Release 21.06.1

The Triton Inference Server container image, release 21.06.1, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.3.1
- cuBLAS 11.5.1.109
- NVIDIA cuDNN 8.2.1
- NVIDIA NCCL 2.9.9 (optimized for NVLink™)
- rdma-core 32.1
- OpenMPI 4.1.1rc1
- OpenUCX 1.10.1
- GDRCopy 2.2
- NVIDIA HPC-X 2.8.2rc3
- TensorRT 7.2.3.4

## Driver Requirements

Release 21.06.1 is based on NVIDIA CUDA 11.3.1, which requires NVIDIA Driver release 465.19.01 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.06.1 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Forest Inference Library (FIL) backend is added to Triton. The FIL backend allows forest models trained by several popular machine learning frameworks (including XGBoost, LightGBM, Scikit-Learn, and cuML) to be deployed in a Triton.
- Windows version of Triton now includes the OpenVino backend.
- The Performance Analyzer (perf_analyzer) now supports testing against the Triton C API.
- The Python backend now allows the use of conda to create a unique execution environment for your Python model. See https://github.com/triton-inference-server/python_backend#using-custom-python-execution-environments.
- Python models that crash or exit unexpectedly are now automatically restarted by Triton.
- Model repositories in S3 storage can now be accessed using HTTPS protocol. See https://github.com/triton-inference-server/server/blob/main/docs/model_repository.md#s3 for more information.
- Triton now collects GPU metrics for MIG partitions.
- Passive model instances can now be specified in the model configuration. A passive model instance will be loaded and initialized by Triton, but no inference requests will be sent to the instance. Passive instances are typically used by a custom backend that uses its own mechanisms to distribute work to the passive instances. See the ModelInstanceGroup section of model_config.proto for the setting.
- NVDLA support is added to the TensorRT backend.
- ONNX Runtime version updated to 1.8.0.
- Windows build documentation simplified and improved.
- Improved detailed and summary reports in Model Analyzer.
- Added an offline mode to Model Analyzer.
- Refer to the 21.06 column of the Frameworks Support Matrix for container image versions that the 21.05 inference server container is based on.
- Ubuntu 20.04 with May 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.06.1 | 2.11.0 | 20.04 | NVIDIA CUDA 11.3.1 | TensorRT 7.2.3.4 |
| 21.06 | | | | |
| 21.05 | 2.10.0 | | NVIDIA CUDA 11.3.0 | |
| 21.04 | 2.9.0 | | | |
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).
- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.
- Compared with the 21.02 and earlier releases, there are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The `utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

# Chapter 9. Triton Inference Server Release 21.06

The Triton Inference Server container image, release 21.06, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.3.1
- cuBLAS 11.5.1.109
- NVIDIA cuDNN 8.2.1
- NVIDIA NCCL 2.9.9 (optimized for NVLink™)
- rdma-core 32.1
- OpenMPI 4.1.1rc1
- OpenUCX 1.10.1
- GDRCopy 2.2
- NVIDIA HPC-X 2.8.2rc3
- TensorRT 7.2.3.4

## Driver Requirements

Release 21.06 is based on NVIDIA CUDA 11.3.1, which requires NVIDIA Driver release 465.19.01 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.06 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Windows version of Triton now includes the OpenVino backend.
- The Performance Analyzer (perf_analyzer) now supports testing against the Triton C API.
- The Python backend now allows the use of conda to create a unique execution environment for your Python model. See https://github.com/triton-inference-server/python_backend#using-custom-python-execution-environments.
- Python models that crash or exit unexpectedly are now automatically restarted by Triton.
- Model repositories in S3 storage can now be accessed using HTTPS protocol. See https://github.com/triton-inference-server/server/blob/main/docs/model_repository.md#s3 for more information.
- Triton now collects GPU metrics for MIG partitions.
- Passive model instances can now be specified in the model configuration. A passive model instance will be loaded and initialized by Triton, but no inference requests will be sent to the instance. Passive instances are typically used by a custom backend that uses its own mechanisms to distribute work to the passive instances. See the ModelInstanceGroup section of model_config.proto for the setting.
- NVDLA support is added to the TensorRT backend.
- ONNX Runtime version updated to 1.8.0.
- Windows build documentation simplified and improved.
- Improved detailed and summary reports in Model Analyzer.
- Added an offline mode to Model Analyzer.
- Refer to the 21.06 column of the Frameworks Support Matrix for container image versions that the 21.05 inference server container is based on.
- Ubuntu 20.04 with May 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.06 | 2.11.0 | 20.04 | NVIDIA CUDA 11.3.1 | TensorRT 7.2.3.4 |
| 21.05 | 2.10.0 | | NVIDIA CUDA 11.3.0 | |
| 21.04 | 2.9.0 | | | |
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event

fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).

- The 21.06 release of Triton was built against the wrong commit of the FIL backend code, causing an incompatible version of RAPIDS to be used instead of the intended RAPIDS 21.06 stable release. This issue is fixed in the new 21.06.1 container released on NGC. Although the Triton server itself and other integrated backends will work, the FIL backend will not work in the 21.06 Triton container
- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.
- Compared with the 21.02 and earlier releases, there are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The `utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

# Chapter 10. Triton Inference Server Release 21.05

The Triton Inference Server container image, release 21.05, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.3.0
- cuBLAS 11.5.1.101
- NVIDIA cuDNN 8.2.0.51
- NVIDIA NCCL 2.9.8 (optimized for NVLink™)
- rdma-core 32.1
- OpenMPI 4.1.1rc1
- OpenUCX 1.10.0
- GDRCopy 2.2
- NVIDIA HPC-X 2.8.2rc3
- TensorRT 7.2.3.4

## Driver Requirements

Release 21.05 is based on NVIDIA CUDA 11.3.0, which requires NVIDIA Driver release 465.19.01 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.05 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Triton on Jetson now supports ONNX via the ONNX Runtime backend.
- The Triton server and HTTP clients (Python and C++) now support compression.
- Ragged batching is now supported for ONNX models.
- The Triton clients have moved to a separate repo: https://github.com/triton-inference-server/client
- Trace now correctly reports all timestamps for all backends.
- NVTX annotations are fixed.
- The legacy custom backend support is removed. All custom backends must be implemented using the TRITONBACKEND API described here: https://github.com/triton-inference-server/backend.
- Added CLI subcommands in Model Analyzer for `profile`, `analyze`, and `report`. See CLI documentation for usage instructions.
  - This is a breaking change and requires updating Model Analyzer config files and CLI flags. See Configuring Model Analyzer and Quick Start for more information.
- Model Analyzer can create a detailed report of any specific model configuration with the `report` subcommand.
- CPU only mode is supported in Model Analyzer.
- Refer to the 21.05 column of the Frameworks Support Matrix for container image versions that the 21.05 inference server container is based on.
- Ubuntu 20.04 with April 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.05 | 2.10.0 | 20.04 | NVIDIA CUDA 11.3.0 | TensorRT 7.2.3.4 |
| 21.04 | 2.9.0 | | | |
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation

occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).

- Running a PyTorch TorchScript model using the PyTorch backend, where multiple instances of a model are configured can lead to a slowdown in model execution due to the following PyTorch issue: https://github.com/pytorch/pytorch/issues/27902.
- Compared with the 21.02 and earlier releases, there are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The `utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

# Chapter 11. Triton Inference Server Release 21.04

The Triton Inference Server container image, release 21.04, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.3.0
- cuBLAS 11.5.1.101
- NVIDIA cuDNN 8.2.0.41
- NVIDIA NCCL 2.9.6 (optimized for NVLink™)
- rdma-core 32.1
- OpenMPI 4.1.1rc1
- OpenUCX 1.10.0
- GDRCopy 2.2
- NVIDIA HPC-X 2.8.2rc3
- TensorRT 7.2.3.4

## Driver Requirements

Release 21.04 is based on NVIDIA CUDA 11.3.0, which requires NVIDIA Driver release 465.19.01 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51 (or later R450), or 460.27 (or later R460). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.04 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Python backend performance has been increased significantly.
- ONNX Runtime update to version 1.7.1.
- Triton Server is now available as a GKE Marketplace Application, see https://github.com/triton-inference-server/server/tree/master/deploy/gke-marketplace-app.
- The GRPC client libraries now allow compression to be enabled.
- Ragged batching is now supported for TensorFlow models.
- For TensorFlow models represented with SavedModel format, it is now possible to choose which graph and signature_def to load. See https://github.com/triton-inference-server/tensorflow_backend/tree/r21.04#parameters.
- A Helm Chart example is added for AWS. See https://github.com/triton-inference-server/server/tree/master/deploy/aws.
- The Model Control API is enhanced to provide an option when unloading an ensemble model. The option allows all contained models to be unloaded as part of unloading the ensemble. See https://github.com/triton-inference-server/server/blob/master/docs/protocol/extension_model_repository.md#model-repository-extension.
- Model reloading using the Model Control API previously resulted in the model being unavailable for a short period of time. This is now fixed so that the model remains available during reloading.
- Latency statistics and metrics for TensorRT models are fixed. Previously the sum of the "compute input", "compute infer" and "compute output" times accurately indicated the entire compute time but the total time could be incorrectly attributed across the three components. This incorrect attribution is now fixed and all values are now accurate.
- Error reporting is improved for the Azure, S3 and GCS cloud file system support.
- Fixed trace support for ensembles. The models contained within an ensemble are now traced correctly.
- Model Analyzer improvements:
  - Summary report now includes GPU Power usage.
  - Model Analyzer will find the Top N model configuration across multiple models.

- Refer to the 21.04 column of the Frameworks Support Matrix for container image versions that the 21.04 inference server container is based on.
- Ubuntu 20.04 with March 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.04 | 2.9.0 | 20.04 | NVIDIA CUDA 11.3.0 | TensorRT 7.2.3.4 |
| 21.03 | 2.8.0 | | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).
- Compared with the 21.02 and earlier releases, there are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The `utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

# Chapter 12. Triton Inference Server Release 21.03

The Triton Inference Server container image, release 21.03, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.2.1 including cuBLAS 11.4.1.1026
- NVIDIA cuDNN 8.1.1
- NVIDIA NCCL 2.8.4 (optimized for NVLink™)
- MLNX_OFED 5.1
- OpenMPI 4.0.5
- TensorRT 7.2.2.3

## Driver Requirements

Release 21.03 is based on NVIDIA CUDA 11.2.1, which requires NVIDIA Driver release 460.32.03 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51(or later R450). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.03 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families.

Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Repository agent is a new extensibility C API added to Triton that allows implementation of custom authentication, decryption, conversion, or similar operations when a model is loaded. See https://github.com/triton-inference-server/server/blob/master/docs/repository_agents.md.
- An OpenVINO backend is added to Triton to enable the execution of OpenVINO models on CPUs. See https://github.com/triton-inference-server/openvino_backend.
- The PyTorch backend is now maintained in its own repository: https://github.com/triton-inference-server/pytorch_backend
- The ONNX Runtime backend is now maintained in its own repository: https://github.com/triton-inference-server/onnxruntime_backend
- The Jetson release of Triton now supports the shared-memory protocol between clients and the Triton server.
- SSL/TLS Mutual Authentication support is added to the GRPC client library.
- A new Model Configuration option, "gather_kernel_buffer_threshold", can be specified to instruct Triton to use a CUDA kernel to gather inputs buffers onto the GPU. Using this option can improve inference performance for some models.
- The Python client libraries have been improved to more efficiently create numpy arrays for input and output tensors.
- The client libraries examples have been improved to more clearly describe how string and byte-blob tensors are supported by the Python Client API. See https://github.com/triton-inference-server/server/blob/master/docs/client_examples.md.
- Ubuntu 20.04 with February 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.03 | 2.8.0 | 20.04 | NVIDIA CUDA 11.2.1 | TensorRT 7.2.2.3 |
| 21.02 | 2.7.0 | | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Triton's TensorRT support depends on the input-consumed feature of TensorRT. In some rare cases using TensorRT 8.0 and earlier versions, the input-consumed event fires earlier than expected, causing Triton to overwrite input tensors while they are still in use and leading to corrupt input data being used for inference. This situation occurs when the inputs feed directly into a TensorRT layer that is optimized into a ForeignNode in the builder log. If you encounter accuracy issues with your TensorRT model, you can work around the issue by enabling the `output_copy_stream` option in your model's configuration (https://github.com/triton-inference-server/common/blob/main/protobuf/model_config.proto#L816).
- There are backwards incompatible changes in the example Python client shared-memory support library when that library is used for tensors of type BYTES. The

`utils.serialize_byte_tensor()` and `utils.deserialize_byte_tensor()` functions now return np.object_ numpy arrays where previously they returned `np.bytes_ numpy` arrays. Code depending on `np.bytes_` must be updated. This change was necessary because the np.bytes_ type removes all trailing zeros from each array element and so binary sequences ending in zero(s) could not be represented with the old behavior. Correct usage of the Python client shared-memory support library is shown inhttps://github.com/triton-inference-server/server/blob/r21.03/src/clients/python/examples/simple_http_shm_string_client.py.

- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 13. Triton Inference Server Release 21.02

The Triton Inference Server container image, release 21.02, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.2.0 including cuBLAS 11.3.1
- NVIDIA cuDNN 8.0.5
- NVIDIA NCCL 2.8.4 (optimized for NVLink™)
- MLNX_OFED 5.1
- OpenMPI 4.0.5
- TensorRT 7.2.2

## Driver Requirements

Release 21.02 is based on NVIDIA CUDA 11.2.0, which requires NVIDIA Driver release 460.27.04 or later. However, if you are running on Data Center GPUs (formerly Tesla), for example, T4, you may use NVIDIA driver release 418.40 (or later R418), 440.33 (or later R440), 450.51(or later R450). The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades and NVIDIA CUDA and Drivers Support.

## GPU Requirements

Release 21.02 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families.

Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Refer to the 21.02 column of the Frameworks Support Matrix for container image versions that the 21.02 inference server container is based on.
- Fixed a bug in TensorRT backend that could, in rare cases, lead to corruption of output tensors.
- Fixed a performance issue in the HTTP/REST client that occurred when the client does not explicitly request specific outputs. In this case all outputs are now returned as binary data where previously they were returned as JSON.
- Added an example Java and Scala client based on GRPC-generated API.
- Extended perf_analyzer to be able to work with TFServing and TorchServe.
- The legacy custom backend API is deprecated and will be removed in a future release. The Triton Backend API should be used as the API for custom backends. The Triton Backend API remains fully supported and that support will continue indefinitely.
- Model Analyzer parameters and test model configurations can be specified with JSON configuration file.
- Model Analyzer will report performance metrics for end-to-end latency and CPU memory usage.
- Ubuntu 20.04 with January 2021 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 21.02 | 2.7.0 | 20.04 | NVIDIA CUDA 11.2.0 | TensorRT 7.2.2.3+cuda11.1.0.024 |
| 20.12 | 2.6.0 | | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.
- Observed memory leak in gRPC client library. Suggested work around process: Restart service to free memory or run within Kubernetes with failover mechanism. For more details on the issue in gRPC, please reference: https://github.com/triton-inference-server/server/issues/2517. The memory leak is fixed on master branch by https://github.com/triton-inference-server/server/pull/2533 and the fix will be included in the 21.03 release. If required, the change can be applied to the 21.02 branch and the client library can be rebuilt: https://github.com/triton-inference-server/server/blob/master/docs/client_libraries.md.

# Chapter 14. Triton Inference Server Release 21.01

The NVIDIA container image release for Triton Inference Server 21.01 has been canceled. The next release will be the 21.02 release which is expected to be released at the end of February.

# Chapter 15. Triton Inference Server Release 20.12

The Triton Inference Server container image, release 20.12, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 20.04 including Python 3.8
- NVIDIA CUDA 11.1.1 including cuBLAS 11.3.0
- NVIDIA cuDNN 8.0.5
- NVIDIA NCCL 2.8.3 (optimized for NVLink™)
- MLNX_OFED 5.1
- OpenMPI 4.0.5
- TensorRT 7.2.2

## Driver Requirements

Release 20.12 is based on NVIDIA CUDA 11.1.1, which requires NVIDIA Driver release 455 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx, 440.30, or 450.xx. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.12 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Refer to the 20.12 column of the Frameworks Support Matrix for container image versions that the 20.12 inference server container is based on.
- Due to interactions with Ubuntu 20.04, the ONNX Runtime's OpenVINO execution provider is disabled in this release. OpenVINO support will be re-enabled in a subsequent release.
- The Triton *-py3-clientsdk container has been renamed to *-py3-sdk and now contains the Model Analyzer as well as the client libraries and examples.
- The PyTorch backend has been moved to a separate repository: https://github.com/triton-inference-server/pytorch_backend. As a result, it is now easy to add or remove it from Triton without requiring a rebuild: see https://github.com/triton-inference-server/server/blob/master/docs/compose.md.
- Initial release of the Model Analyzer tool in the Triton SDK container and PIP package in the NVIDIA Py Index.
- Ubuntu 20.04 with November 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.12 | 2.6.0 | 20.04 | NVIDIA CUDA 11.1.1 | TensorRT 7.2.2 |
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 16. Triton Inference Server Release 20.11

The Triton Inference Server container image, release 20.11, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 11.1.0 including cuBLAS 11.2.1
- NVIDIA cuDNN 8.0.4
- NVIDIA NCCL 2.8.2 (optimized for NVLink™)
- MLNX_OFED 5.1
- OpenMPI 4.0.5
- TensorRT 7.2.1

## Driver Requirements

Release 20.11 is based on NVIDIA CUDA 11.1.0, which requires NVIDIA Driver release 455 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx, 440.30, or 450.xx. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.11 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- ONNX Runtime backend updated to use ONNX Runtime 1.5.3.
- The PyTorch backend is moved to a dedicated repo: triton-inference-server/pytorch_backend.
- The Caffe2 backend is removed. Caffe2 models are no longer supported.
- Fixed handling of failed model reloads. If a model reload fails, the currently loaded version of the model will remain loaded and its availability will be uninterrupted.
- Releasing Triton ModelAnalyzer in the Triton SDK container and as a PIP package available in NVIDIA PyIndex.
- Refer to the 20.11 column of the Frameworks Support Matrix for container image versions that the 20.11 inference server container is based on.
- Ubuntu 18.04 with October 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.11 | 2.5.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.10 | 2.4.0 | | | |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 17. Triton Inference Server Release 20.10

The Triton Inference Server container image, release 20.10, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 11.1.0 including cuBLAS 11.2.1
- NVIDIA cuDNN 8.0.4
- NVIDIA NCCL 2.7.8 (optimized for NVLink™)
- MLNX_OFED
- OpenMPI 3.1.6
- TensorRT 7.2.1

## Driver Requirements

Release 20.10 is based on NVIDIA CUDA 11.1.0, which requires NVIDIA Driver release 455 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx, 440.30, or 450.xx. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.10 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- A new Python backend allows Python code to run as a model within Triton. See https://github.com/triton-inference-server/python_backend.
- A new DALI backend allows running pre-processing and augmentation pipelines within Triton. See https://github.com/triton-inference-server/dali_backend.
- The perf_client application is renamed to perf_analyzer; functionality remains the same.
- A new Model Analyzer project is started with the goal of providing analysis and guidance on how to best optimize single or multiple models within Triton. The initial release analyzes GPU memory usage. See https://github.com/triton-inference-server/model_analyzer.
- Triton documentation now resides on GitHub and is reachable from https://github.com/triton-inference-server/server/blob/master/README.md.
- Build process for Triton has changed, see https://github.com/triton-inference-server/server/blob/master/docs/build.md.
- Triton backends are moving to separate repositories. In this release the TensorFlow, ONNX Runtime, Python and DALI backends are moved; see https://github.com/triton-inference-server/backend#where-can-i-find-all-the-backends-that-are-available-for-triton.
- Refer to the 20.10 column of the Frameworks Support Matrix for container image versions that the 20.10 inference server container is based on.
- Ubuntu 18.04 with September 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.10 | 2.4.0 | 18.04 | NVIDIA CUDA 11.1.0 | TensorRT 7.2.1 |
| 20.09 | 2.3.0 | | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 18. Triton Inference Server Release 20.09

The Triton Inference Server container image, release 20.09, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 11.0.3 including cuBLAS 11.2.0
- NVIDIA cuDNN 8.0.4
- NVIDIA NCCL 2.7.8 (optimized for NVLink™)
- MLNX_OFED
- OpenMPI 3.1.6
- TensorRT 7.1.3

## Driver Requirements

Release 20.09 is based on NVIDIA CUDA 11.0.3, which requires NVIDIA Driver release 450 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.09 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Python Client library is now a pip package available from the NVIDIA pypi index. See https://github.com/triton-inference-server/server/blob/master/src/clients/python/library/README.md for more information.
- Fixed a performance issue with the HTTP/REST protocol and the Python client library that caused reduced performance when outputs were not requested explicitly in an inference request.
- Fixed some issues in reporting of statistics for ensemble models.
- GRPC updated to version 1.25.0.
- Refer to the 20.09 column of the Frameworks Support Matrix for container image versions that the 20.09 inference server container is based on.
- Ubuntu 18.04 with August 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.09 | 2.3.0 | 18.04 | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.08 | 2.2.0 | | | |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- The KFServing HTTP/REST and GRPC protocols and corresponding Python and C++ clients are beta quality and are likely to change.
- The new C API specified in `tritonserver.h` is beta quality and is likely to change.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 19. Triton Inference Server Release 20.08

The Triton Inference Server container image, release 20.08, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 11.0.3 including cuBLAS 11.2.0
- NVIDIA cuDNN 8.0.2
- NVIDIA NCCL 2.7.8 (optimized for NVLink™)
- MLNX_OFED
- OpenMPI 3.1.6
- TensorRT 7.1.3

## Driver Requirements

Release 20.08 is based on NVIDIA CUDA 11.0.3, which requires NVIDIA Driver release 450 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.08 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- TensorFlow 2.x is now supported in addition to TensorFlow 1.x. See the Frameworks Support Matrix for the supported TensorFlow versions. The version of TensorFlow used can be selected when launching Triton with the `--backend-config=tensorflow,version=<version>` flag. Set <version> to 1 or 2 to select TensorFlow1 or TensorFlow2 respectively. By default, TensorFlow1 is used.
- Added inference request timeout option to Python and C++ client libraries.
- Updated GRPC inference protocolti to fix performance regression.
- Explicit major/minor versioning added to TRITONSERVER and TRITONBACKED APIs.
- New CMake option TRITON_CLIENT_SKIP_EXAMPLES to disable building the client examples.
- Refer to the 20.08 column of the Frameworks Support Matrix for container image versions that the 20.08 inference server container is based on.
- Ubuntu 18.04 with July 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.08 | 2.2.0 | 18.04 | NVIDIA CUDA 11.0.3 | TensorRT 7.1.3 |
| 20.07 | 1.15.0<br>2.1.0 | | NVIDIA CUDA 11.0.194 | |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| | 1.8.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.11 | | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- The KFServing HTTP/REST and GRPC protocols and corresponding Python and C++ clients are beta quality and are likely to change.
- The new C API specified in `tritonserver.h` is beta quality and is likely to change.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 20. Triton Inference Server Release 20.07

The Triton Inference Server container image, release 20.07, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 11.0.194 including cuBLAS 11.1.0
- NVIDIA cuDNN 8.0.1
- NVIDIA NCCL 2.7.6 (optimized for NVLink™)
- MLNX_OFED
- OpenMPI 3.1.6
- TensorRT 7.1.3

## Driver Requirements

Release 20.07 is based on NVIDIA CUDA 11.0.194, which requires NVIDIA Driver release 450 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.07 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- For Triton V2, add TensorFlow optimization option that enables automatic FP16 optimization of the model.
- For Triton V2, the PyTorch backend now includes support for TorchVision operations.
- This release includes support for both the new KFServing based protocols as well as the legacy V1 protocols.
- Support for the new KFServing HTTP/REST, GRPC and corresponding client libraries is released on GitHub branch r20.07 and as NGC container 20.07-py3.
- Support for the legacy V1 HTTP/REST, GRPC and corresponding client libraries is released on GitHub branch r20.07-v1 and as NGC container 20.07-v1-py3.
- Migration from Triton V1 to Triton V2 requires significant changes; see the "Backwards Compatibility" and "Roadmap" sections of the GitHub README for more information.
- Refer to the 20.07 column of the Frameworks Support Matrix for container image versions that the 20.07 inference server container is based on.
- Ubuntu 18.04 with June 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.07 | 1.15.0<br>2.1.0 | 18.04 | NVIDIA CUDA 11.0.194 | TensorRT 7.1.3 |
| 20.06 | 1.14.0<br>2.0.0 | | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- When using the TensorRT NGC container to generate TensorRT models for Triton, the 20.07.1 version of the TensorRT container must be used to ensure compatibility with Triton 20.07.
- The KFServing HTTP/REST and GRPC protocols and corresponding Python and C++ clients are beta quality and are likely to change.
- The new C API specified in `tritonserver.h` is beta quality and is likely to change.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 21. Triton Inference Server Release 20.06

The Triton Inference Server container image, release 20.06, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 11.0.167 including cuBLAS 11.1.0
- NVIDIA cuDNN 8.0.1
- NVIDIA NCCL 2.7.5 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.6
- TensorRT 7.1.2

## Driver Requirements

Release 20.06 is based on NVIDIA CUDA 11.0.167, which requires NVIDIA Driver release 450 or later. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.06 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, Turing, and Ampere Architecture GPU families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Updates for KFserving HTTP/REST and GRPC protocols and corresponding Python and C++ client libraries. This release includes support for both the new KFServing based protocols as well as the legacy V1 protocols.
- Support for the new KFServing HTTP/REST, GRPC and corresponding client libraries is released on GitHub branch r20.06 and as NGC container 20.06-py3.
- Support for the legacy V1 HTTP/REST, GRPC and corresponding client libraries is released on GitHub branch r20.06-v1 and as NGC container 20.06-v1-py3.
- Migration from Triton V1 to Triton V2 requires significant changes; see the "Backwards Compatibility" and "Roadmap" sections of the GitHub README for more information.
- Refer to the 20.06 column of the Frameworks Support Matrix for container image versions that the 20.06 inference server container is based on.
- The latest version of NVIDIA CUDA 11.0.167 including cuBLAS 11.1.0
- The latest version of NVIDIA cuDNN 8.0.1
- The latest version of NVIDIA NCCL 2.7.5
- The latest version of OpenMPI 3.1.6
- The latest version of TensorRT 7.1.2
- Ubuntu 18.04 with May 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.06 | 1.14.0<br>2.0.0 | 18.04 | NVIDIA CUDA 11.0.167 | TensorRT 7.1.2 |
| 20.03.1 | 1.13.0 | | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.03 | 1.12.0 | | | |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- The KFServing HTTP/REST and GRPC protocols and corresponding V2 experimental Python and C++ clients are beta quality and are likely to change.
- The new C API specified in `tritonserver.h` is beta quality and is likely to change.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 22. Triton Inference Server Release 20.03.1

The Triton Inference Server container image, release 20.03.1, is available on NGC and is open source on GitHub.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- NVIDIA cuDNN 7.6.5
- NVIDIA NCCL 2.6.3 (optimized for NVLink™)
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 7.0.0

## Driver Requirements

Release 20.03.1 is based on NVIDIA CUDA 10.2.89, which requires NVIDIA Driver release 440.33.01. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+, 410, 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.03.1 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the NVIDIA Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Updates for KFserving HTTP/REST and GRPC protocols and corresponding Python and C++ client libraries. See the Roadmap section of the README for more information.
- Updated GRPC version to 1.24.0.
- Several issues with S3 storage were resolved.
- Fixed `last_inferrence_timestamp` value to correctly show the time when inference last occurred for each model.
- The Caffe2 backend is deprecated. Support for Caffe2 models will be removed in a future release.
- Refer to the 20.03 column of the Frameworks Support Matrix for container image versions that the 20.03.1 inference server container is based on.
- The inference server container image version 20.03.1 is additionally based on ONNX Runtime 1.2.0.
- Ubuntu 18.04 with April 2020 updates.

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

<table>
<tr><th>Container Version</th><th>Triton Inference Server</th><th>Ubuntu</th><th>CUDA Toolkit</th><th>TensorRT</th></tr>
<tr><td>20.03.1</td><td>1.13.0</td><td rowspan="10">18.04</td><td rowspan="6">NVIDIA CUDA 10.2.89</td><td rowspan="3">TensorRT 7.0.0</td></tr>
<tr><td>20.03</td><td>1.12.0</td></tr>
<tr><td>20.02</td><td>1.11.0</td></tr>
<tr><td>20.01</td><td>1.10.0</td></tr>
<tr><td>19.12</td><td>1.9.0</td><td rowspan="3">TensorRT 6.0.1</td></tr>
<tr><td>19.11</td><td>1.8.0</td></tr>
<tr><td>19.10</td><td>1.7.0</td><td rowspan="3">NVIDIA CUDA 10.1.243</td></tr>
<tr><td>19.09</td><td>1.6.0</td></tr>
<tr><td>19.08</td><td>1.5.0</td><td>TensorRT 5.1.5</td></tr>
</table>

## Known Issues

- The KFServing HTTP/REST and GRPC protocols and corresponding V2 experimental Python and C++ clients are beta quality and are likely to change. Specifically:
  - The data returned by the statistics API will be changing to include additional information.
  - The data returned by the repository index API will be changing to include additional information.
- The new C API specified in `tritonserver.h` is beta quality and is likely to change.
- When using the experimental V2 HTTP/REST C++ client, classification results are not supported for output tensors.
- When using the experimental V2 perf_client_v2, for high concurrency values perf_client_v2 may not be able to achieve throughput as high as V1 perf_client.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 23. Triton Inference Server Release 20.03

The Triton Inference Server container image, release 20.03, is available on NGC and is open source on GitHub.

Starting in release 20.03, TensorRT Inference Server is now called Triton Inference Server.

## Contents of the Triton Inference Server container

The Triton Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tritonserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- NVIDIA cuDNN 7.6.5
- NVIDIA NCCL 2.6.3 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 7.0.0

## Driver Requirements

Release 20.03 is based on NVIDIA CUDA 10.2.89, which requires NVIDIA Driver release 440.33.01. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+, 410, 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.03 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Added queuing policies for dynamic batching scheduler. These policies are specified in the model configuration and allow each model to set maximum queue size, time outs, and priority levels for inference requests.
- Support for large ONNX models where weights are stored in separate files.
- Allow ONNX Runtime optimization level to be configured via the model configuration optimization setting.
- Experimental Python client and server support for community standard GRPC inferencing API.
- Added `--min-supported-compute-capability` flag to allow Triton Server to use older, unsupported GPUs.
- Fixed `perf_client` shared memory support. In some cases the shared-memory option did not work correctly due to the input and output tensor names. This issue is now resolved.
- Refer to the Frameworks Support Matrix for container image versions that the 20.03 inference server container is based on.
- The inference server container image version 20.03 is additionally based on ONNX Runtime 1.1.1.
- Ubuntu 18.04 with February 2020 updates

## NVIDIA Triton Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, Triton Inference Server, and TensorRT are supported in each of the NVIDIA containers for Triton Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 20.03 | 1.12.0 | 18.04<br>16.04 | NVIDIA CUDA 10.2.89 | TensorRT 7.0.0 |
| 20.02 | 1.11.0 | | | |
| 20.01 | 1.10.0 | | | |

| Container Version | Triton Inference Server | Ubuntu | CUDA Toolkit | TensorRT |
|---|---|---|---|---|
| 19.12 | 1.9.0 | | | TensorRT 6.0.1 |
| 19.11 | 1.8.0 | | | |
| 19.10 | 1.7.0 | | NVIDIA CUDA 10.1.243 | |
| 19.09 | 1.6.0 | | | |
| 19.08 | 1.5.0 | | | TensorRT 5.1.5 |

## Known Issues

- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 24. Triton Inference Server Release 20.02

The TensorRT Inference Server container image, release 20.02, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- NVIDIA cuDNN 7.6.5
- NVIDIA NCCL 2.5.6 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 7.0.0

## Driver Requirements

Release 20.02 is based on NVIDIA CUDA 10.2.89, which requires NVIDIA Driver release 440.33.01. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+, 410, 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.02 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Refer to the Frameworks Support Matrix for container image versions that the 20.02 inference server container is based on.
- The inference server container image version 20.02 is additionally based on ONNX Runtime 1.1.1.
- The TensorRT backend is improved to have significantly better performance. Improvements include reducing thread contention, using pinned memory for faster CPU<->GPU transfers, and increasing compute and memory copy overlap on GPUs.
- Reduce memory usage of TensorRT models in many cases by sharing weights across multiple model instances.
- Boolean data-type and shape tensors are now supported for TensorRT models.
- A new model configuration option allows the dynamic batcher to create "ragged" batches for custom backend models. A ragged batch is a batch where one or more of the input/output tensors have different shapes in different batch entries.
- Local S3 storage endpoints are now supported for model repositories. A local S3 endpoint is specified as `s3://host:port/path/to/repository`.
- The Helm chart showing an example Kubernetes deployment is updated to include Prometheus and Grafana support so that inference server metrics can be collected and visualized.
- The inference server container no longer sets LD_LIBRARY_PATH, instead the server uses RUNPATH to locate its shared libraries.
- Python 2 is end-of-life so all support has been removed. Python 3 is still supported.
- Ubuntu 18.04 with January 2020 updates

## NVIDIA TensorRT Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, TensorRT Inference Server, and TensorRT are supported in each of the NVIDIA containers for TensorRT Inference Server. For older container versions, refer to the Frameworks Support Matrix.

| Container Version | Ubuntu | CUDA Toolkit | TensorRT Inference Server | TensorRT |
|---|---|---|---|---|
| 20.02 | 18.04 | NVIDIA CUDA 10.2.89 | 1.12.0 | TensorRT 7.0.0 |
| 20.01 | | | 1.11.0 | |
| | | | 1.10.0 | |
| 19.12 | | | 1.9.0 | TensorRT 6.0.1 |
| | | | 1.8.0 | |

| Container Version | Ubuntu | CUDA Toolkit | TensorRT Inference Server | TensorRT |
|---|---|---|---|---|
| 19.11 | | | | |
| 19.10 | | NVIDIA CUDA 10.1.243 | 1.7.0 | |
| 19.09 | | | 1.6.0 | |
| 19.08 | | | 1.5.0 | TensorRT 5.1.5 |

## Known Issues

- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 25. Triton Inference Server Release 20.01

The TensorRT Inference Server container image, release 20.01, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- NVIDIA cuDNN 7.6.5
- NVIDIA NCCL 2.5.6 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 7.0.0

## Driver Requirements

Release 20.01 is based on NVIDIA CUDA 10.2.89, which requires NVIDIA Driver release 440.33.01. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+, 410, 418.xx or 440.30. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 20.01 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Refer to the Frameworks Support Matrix for container image versions that the 20.01 inference server container is based on.
- The inference server container image version 20.01 is additionally based on ONNX Runtime 1.1.1.
- Server status can be requested in JSON format using the HTTP/REST API. Use endpoint `/api/status?format=json`.
- The dynamic batcher now has an option to preserve the ordering of batched requests when there are multiple model instances. See model_config.proto for more information.
- Latest version of TensorRT 7.0.0
- Ubuntu 18.04 with December 2019 updates

## NVIDIA TensorRT Inference Server Container Versions

The following table shows what versions of Ubuntu, CUDA, TensorRT Inference Server, and TensorRT are supported in each of the NVIDIA containers for TensorRT Inference Server. For older container versions, refer to the Frameworks Support Matrix.

<table>
<tr><th>Container Version</th><th>Ubuntu</th><th>CUDA Toolkit</th><th>TensorRT Inference Server</th><th>TensorRT</th></tr>
<tr><td>20.01</td><td rowspan="6">18.04<br>16.04</td><td rowspan="3">NVIDIA CUDA 10.2.89</td><td>1.10.0</td><td>TensorRT 7.0.0</td></tr>
<tr><td>19.12</td><td>1.9.0</td><td rowspan="4">TensorRT 6.0.1</td></tr>
<tr><td>19.11</td><td>1.8.0</td></tr>
<tr><td>19.10</td><td rowspan="2">NVIDIA CUDA 10.1.243</td><td>1.7.0</td></tr>
<tr><td>19.09</td><td>1.6.0</td></tr>
<tr><td>19.08</td><td>1.5.0</td><td>TensorRT 5.1.5</td></tr>
</table>

## Known Issues

- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 26. Triton Inference Server Release 19.12

The TensorRT Inference Server container image, release 19.12, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- NVIDIA cuDNN 7.6.5
- NVIDIA NCCL 2.5.6 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 6.0.1

## Driver Requirements

Release 19.12 is based on NVIDIA CUDA 10.2.89, which requires NVIDIA Driver release 440.30. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+, 410, 418.xx or 440.30.. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.12 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Refer to the Frameworks Support Matrix for container image versions that the 19.12 inference server container is based on.
- The inference server container image version 19.12 is additionally based on ONNX Runtime 1.1.1.
- The model configuration now includes a *model warmup* option. This option provides the ability to tune and optimize the model before inference requests are received, avoiding initial inference delays. This option is especially useful for frameworks like TensorFlow that perform network optimization in response to the initial inference requests. Models can be warmed-up with one or more synthetic or realistic workloads before they become ready in the server
- An enhanced sequence batcher now has multiple scheduling strategies. A new *Oldest* strategy integrates with the dynamic batcher to enable improved inference performance for models that don't require all inference requests in a sequence to be routed to the same batch slot.
- The `perf_client` now has an option to generate requests using a realistic poisson distribution or a user provided distribution.
- A new repository API (available in the shared library API, HTTP, and gRPC) returns an index of all models available in the model repositories) visible to the server. This index can be used to see what models are available for loading onto the server.
- The server status returned by the server status API now includes the timestamp of the last inference request received for each model.
- Inference server tracing capabilities are now documented in the Optimization section of the *User Guide*. Tracing support is enhanced to provide trace for ensembles and the contained models.
- A community contributed Dockerfile is now available to build the TensorRT Inference Server clients on CentOS.
- Ubuntu 18.04 with November2019 updates

## Known Issues

- The beta of the custom backend API version 2 has non-backwards compatible changes to enable complete support for input and output tensors in both CPU and GPU memory:
  - The signature of the `CustomGetNextInputV2Fn_t` function adds the `memory_type_id` argument.
  - The signature of the `CustomGetOutputV2Fn_t` function adds the `memory_type_id` argument.

- The beta of the inference server library API has non-backwards compatible changes to enable complete support for input and output tensors in both CPU and GPU memory:
  - The signature and operation of the `TRTSERVER_ResponseAllocatorAllocFn_t` function has changed. See `src/core/trtserver.h` for a description of the new behavior.
  - The signature of the `TRTSERVER_InferenceRequestProviderSetInputData` function adds the `memory_type_id` argument.
  - The signature of the `TRTSERVER_InferenceResponseOutputData` function add the `memory_type_id` argument.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 27. Triton Inference Server Release 19.11

The TensorRT Inference Server container image, release 19.11, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04 including Python 3.6
- NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- NVIDIA cuDNN 7.6.5
- NVIDIA NCCL 2.5.6 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 6.0.1

## Driver Requirements

Release 19.11 is based on NVIDIA CUDA 10.2.89, which requires NVIDIA Driver release 440.30. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+, 410 or 418.xx. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.11 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- Refer to the Frameworks Support Matrix for container image versions that the 19.11 inference server container is based on.
- The inference server container image version 19.11 is additionally based on ONNX Runtime 1.1.1.
- Shared-memory support is expanded to include CUDA shared memory.
- Improve efficiency of pinned-memory used for ensemble models.
- The `perf_client` application has been improved with easier-to-use command-line arguments (while maintaining compatibility with existing arguments).
- Support for string tensors added to `perf_client`.
- Documentation contains a new *Optimization* section discussing some common optimization strategies and how to use `perf_client` to explore these strategies.
- Latest version of NVIDIA CUDA 10.2.89 including cuBLAS 10.2.2.89
- Latest version of NVIDIA cuDNN 7.6.5
- Latest version of NVIDIA NCCL 2.5.6
- Ubuntu 18.04 with October 2019 updates

## Deprecated Features

- The asynchronous inference API has been modified in the C++ and Python client libraries.
  - In the C++ library:
    - The non-callback version of the `AsyncRun` function is removed.
    - The `GetReadyAsyncRequest` function is removed.
    - The signature of the `GetAsyncRunResults` function was changed to remove the `is_ready` and `wait` arguments.
  - In the Python library:
    - The non-callback version of the `async_run` function was removed.
    - The `get_ready_async_request` function was removed.
    - The signature of the `get_async_run_results` function was changed to remove the `wait` argument.

## Known Issues

- The beta of the custom backend API version 2 has non-backwards compatible changes to enable complete support for input and output tensors in both CPU and GPU memory:

- The signature of the `CustomGetNextInputV2Fn_t` function adds the `memory_type_id` argument.
- The signature of the `CustomGetOutputV2Fn_t` function adds the `memory_type_id` argument.

- The beta of the inference server library API has non-backwards compatible changes to enable complete support for input and output tensors in both CPU and GPU memory:
  - The signature and operation of the `TRTSERVER_ResponseAllocatorAllocFn_t` function has changed. See `src/core/trtserver.h` for a description of the new behavior.
  - The signature of the `TRTSERVER_InferenceRequestProviderSetInputData` function adds the `memory_type_id` argument.
  - The signature of the `TRTSERVER_InferenceResponseOutputData` function add the `memory_type_id` argument.
- TensorRT reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 28. Triton Inference Server Release 19.10

The TensorRT Inference Server container image, release 19.10, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04
- NVIDIA CUDA 10.1.243 including cuBLAS 10.2.1.243
- NVIDIA cuDNN 7.6.4
- NVIDIA NCCL 2.4.8 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 6.0.1

## Driver Requirements

Release 19.10 is based on NVIDIA CUDA 10.1.243, which requires NVIDIA Driver release 418.xx. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.10 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.10 is based on NVIDIA TensorRT Inference Server 1.7.0, TensorFlow 1.14.0, ONNX Runtime 0.4.0, and PyTorch 1.3.0.
- A Client SDK container is now provided on NGC in addition to the inference server container. The client SDK container includes the client libraries and examples.
- Latest version of NVIDIA cuDNN 7.6.4
- TensorRT optimization may now be enabled for any TensorFlow model by enabling the feature in the optimization section of the model configuration.
- The ONNXRuntime backend now includes the TensorRT and Open Vino execution providers. These providers are enabled in the optimization section of the model configuration.
- Automatic configuration generation (`--strict-model-config=false`) now works correctly for TensorRT models with variable-sized inputs and/or outputs.
- Multiple model repositories may now be specified on the command line. Optional command-line options can be used to explicitly load specific models from each repository.
- Ensemble models are now pruned dynamically so that only models needed to calculate the requested outputs are executed.
- The example clients now include a simple Go example that uses the GRPC API.
- Ubuntu 18.04 with September 2019 updates

## Known Issues

- In TensorRT 6.0.1, reformat-free I/O is not supported.
- Some versions of Google Kubernetes Engine (GKE) contain a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version or a GKE 1.14.6 or later version to avoid this issue.

# Chapter 29. Triton Inference Server Release 19.09

The TensorRT Inference Server container image, release 19.09, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04
- NVIDIA CUDA 10.1.243 including cuBLAS 10.2.1.243
- NVIDIA cuDNN 7.6.3
- NVIDIA NCCL 2.4.8 (optimized for NVLink™ )
- MLNX_OFED
- OpenMPI 3.1.4
- TensorRT 6.0.1

## Driver Requirements

Release 19.09 is based on NVIDIA CUDA 10.1.243, which requires NVIDIA Driver release 418.xx. However, if you are running on Tesla (for example, T4 or any other Tesla board), you may use NVIDIA driver release 396, 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.09 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.09 is based on NVIDIA TensorRT Inference Server 1.6.0, TensorFlow 1.14.0, ONNX Runtime 0.4.0, and PyTorch 1.2.0.
- Latest version of NVIDIA cuDNN 7.6.3
- Latest version of TensorRT 6.0.1
- Added TensorRT 6 support, which includes support for TensorRT dynamic shapes
- Shared memory support is added as an alpha feature in this release. This support allows input and output tensors to be communicated via shared memory instead of over the network. Currently only system (CPU) shared memory is supported.
- Amazon S3 is now supported as a remote file system for model repositories. Use the `s3://` prefix on model repository paths to reference S3 locations.
- The inference server library API is available as a beta in this release. The library API allows you to link against `libtrtserver.so` so that you can include all the inference server functionality directly in your application.
- GRPC endpoint performance improvement. The inference server's GRPC endpoint now uses significantly less memory while delivering higher performance.
- The ensemble scheduler is now more flexible in allowing batching and non-batching models to be composed together in an ensemble.
- The ensemble scheduler will now keep tensors in GPU memory between models when possible. Doing so significantly increases performance of some ensembles by avoiding copies to and from system memory.
- The performance client, `perf_client`, now supports models with variable-sized input tensors.
- Ubuntu 18.04 with August 2019 updates

## Known Issues

- The ONNX Runtime backend could not be updated to the 0.5.0 release due to multiple performance and correctness issues with that release.
- TensorRT 6:
  - Reformat-free I/O is not supported.
  - Only models that have a single optimization profile are currently supported.
- Google Kubernetes Engine (GKE) version 1.14 contains a regression in the handling of `LD_LIBRARY_PATH` that prevents the inference server container from running correctly (see issue 141255952). Use a GKE 1.13 or earlier version to avoid this issue.

# Chapter 30. Triton Inference Server Release 19.08

The TensorRT Inference Server container image, release 19.08, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04
- NVIDIA CUDA 10.1.243 including cuBLAS 10.2.1.243
- NVIDIA cuDNN 7.6.2
- NVIDIA NCCL 2.4.8 (optimized for NVLink™ )
- MLNX_OFED +4.0
- OpenMPI 3.1.4
- TensorRT 5.1.5

## Driver Requirements

Release 19.08 is based on NVIDIA CUDA 10.1.243, which requires NVIDIA Driver release 418.87. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.08 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.08 is based on NVIDIA TensorRT Inference Server 1.5.0, TensorFlow 1.14.0, ONNX Runtime 0.4.0, and PyTorch 1.2.0a0.
- Added a new execution mode allows the inference server to start without loading any models from the model repository. Model loading and unloading is then controlled by a new GRPC/HTTP model control API.
- Added a new instance-group mode allows TensorFlow models that explicitly distribute inferencing across multiple GPUs to run in that manner in the inference server.
- Improved input/output tensor reshape to allow variable-sized dimensions in tensors being reshaped.
- Added a C++ wrapper around the custom backend C API to simplify the creation of custom backends. This wrapper is included in the custom backend SDK.
- Improved the accuracy of the *compute* statistic reported for inference requests. Previously the compute statistic included some additional time beyond the actual compute time.
- The performance client, `perf_client`, now reports more information for ensemble models, including statistics for all contained models and the entire ensemble.
- Latest version of NVIDIA CUDA 10.1.243 including cuBLAS 10.2.1.243
- Latest version of NVIDIA cuDNN 7.6.2
- Latest version of NVIDIA NCCL 2.4.8
- Latest version of MLNX_OFED +4.0
- Latest version of OpenMPI 3.1.4
- Ubuntu 18.04 with July 2019 updates

## Known Issues

There are no known issues in this release.

# Chapter 31. Triton Inference Server Release 19.07

The TensorRT Inference Server container image, release 19.07, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 18.04
- NVIDIA CUDA 10.1.168 including cuBLAS 10.2.0.168
- NVIDIA cuDNN 7.6.1
- NVIDIA NCCL 2.4.7 (optimized for NVLink™ )
- MLNX_OFED +3.4
- OpenMPI 3.1.3
- TensorRT 5.1.5

## Driver Requirements

Release 19.07 is based on NVIDIA CUDA 10.1.168, which requires NVIDIA Driver release 418.67. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.07 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.07 is based on NVIDIA TensorRT Inference Server 1.4.0, TensorFlow 1.14.0, ONNX Runtime 0.4.0, and Caffe2 0.8.2.
- Added `libtorch` as a new backend. PyTorch models manually decorated or automatically traced to produce TorchScript can now be run directly by the inference server.
- Build system converted from bazel to CMake. The new CMake-based build system is more transparent, portable and modular.
- To simplify the creation of custom backends, a Custom Backend SDK and improved documentation is now available.
- Improved AsyncRun API in C++ and Python client libraries.
- `perf_client` can now use user-supplied input data (previously used random or zero input data).
- `perf_client` now reports latency at multiple confidence percentiles (p50, p90, p95, p99) as well as a user-supplied percentile that is also used to stabilize latency results.
- Improvements to automatic model configuration creation (`--strict-model-config=false`).
- C++ and Python client libraries now allow additional HTTP headers to be specified when using the HTTP protocol.
- Latest version of NVIDIA cuDNN 7.6.1
- Latest version of MLNX_OFED +3.4
- Latest version of Ubuntu 18.04

## Known Issues

There are no known issues in this release.

# Chapter 32. Triton Inference Server Release 19.06

The TensorRT Inference Server container image, release 19.06, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04
- NVIDIA CUDA 10.1.168 including cuBLAS 10.2.0.168
- NVIDIA cuDNN 7.6.0
- NVIDIA NCCL 2.4.7 (optimized for NVLink™)
- OpenMPI 3.1.3
- TensorRT 5.1.5

## Driver Requirements

Release 19.06 is based on NVIDIA CUDA 10.1.168, which requires NVIDIA Driver release 418.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.06 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.06 is based on NVIDIA TensorRT Inference Server 1.3.0, TensorFlow 1.13.1, ONNX Runtime 0.4.0, and Caffe2 0.8.2.
- Latest version of NVIDIA CUDA 10.1.168 including cuBLAS 10.2.0.168
- Latest version of NVIDIA NCCL 2.4.7
- Added ONNX Runtime as a new backend. The ONNX Runtime backend allows the inference server to directly run ONNX models without requiring conversion to Caffe2 or TensorRT.
- HTTP health port may be specified independently of inference and status HTTP port with `--http-health-port` flag.
- Fixed bug in `perf_client` that caused high CPU usage by client that could lower the measured inference/sec in some cases.
- Ubuntu 16.04 with May 2019 updates (see Announcements)

## Announcements

In the next release, we will no longer support Ubuntu 16.04. Release 19.07 will instead support Ubuntu 18.04.

## Known Issues

- Google Cloud Storage (GCS) support is not available in this release. Support for GCS will be re-enabled in the 19.07 release.

# Chapter 33. Triton Inference Server Release 19.05

The TensorRT Inference Server container image, release 19.05, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04
- NVIDIA CUDA 10.1 Update 1 including cuBLAS 10.1 Update 1
- NVIDIA cuDNN 7.6.0
- NVIDIA NCCL 2.4.6 (optimized for NVLink™)
- OpenMPI 3.1.3
- TensorRT 5.1.5

## Driver Requirements

Release 19.05 is based on CUDA 10.1 Update 1, which requires NVIDIA Driver release 418.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.05 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.05 is based on NVIDIA TensorRT Inference Server 1.2.0, TensorFlow 1.13.1, and Caffe2 0.8.2.
- Latest version of NVIDIA CUDA 10.1 Update 1 including cuBLAS 10.1 Update 1
- Latest version of NVIDIA cuDNN 7.6.0
- Latest version of TensorRT 5.1.5
- Ensembling is now available. An ensemble represents a pipeline of one or more models and the connection of input and output tensors between those models. A single inference request to an ensemble will trigger the execution of the entire pipeline.
- Added a Helm chart that deploys a single TensorRT Inference Server into a Kubernetes cluster.
- The client `Makefile` now supports building for both Ubuntu 18.04 and Ubuntu 16.04. The Python wheel produced from the build is now compatible with both Python2 and Python3.
- The `perf_client` application now has a `--percentile` flag that can be used to report latencies instead of reporting average latency (which remains the default). For example, using `--percentile=99` causes `perf_client` to report the 99th percentile latency.
- The `perf_client` application now has a `-z` option to use zero-valued input tensors instead of random values.
- Improved error reporting of incorrect input/output tensor names for TensorRT models.
- Added `--allow-gpu-metrics` option to enable/disable reporting of GPU metrics.

## Known Issues

There are no known issues in this release.

# Chapter 34. Triton Inference Server Release 19.04

The TensorRT Inference Server container image, release 19.04, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04
- NVIDIA CUDA 10.1.105 including cuBLAS 10.1.0.105
- NVIDIA cuDNN 7.5.0
- NVIDIA NCCL 2.4.6 (optimized for NVLink™)
- OpenMPI 3.1.3
- TensorRT 5.1.2 RC

## Driver Requirements

Release 19.04 is based on CUDA 10.1, which requires NVIDIA Driver release 418.xx.x+. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.04 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.04 is based on NVIDIA TensorRT Inference Server 1.1.0, TensorFlow 1.13.1, and Caffe2 0.8.2.
- Latest version of NVIDIA NCCL 2.4.6
- Latest version of cuBLAS 10.1.0.105
- Client libraries and examples now build with a separate Makefile (a Dockerfile is also included for convenience).
- Input or output tensors with variable-size dimensions (indicated by -1 in the model configuration) can now represent tensors where the variable dimension has value 0 (zero).
- Zero-sized input and output tensors are now supported for batching models. This enables the inference server to support models that require inputs and outputs that have shape `[ batch-size ]`.
- TensorFlow custom operations (C++) can now be built into the inference server. An example and documentation are included in this release.
- Ubuntu 16.04 with March 2019 updates

## Known Issues

There are no known issues in this release.

# Chapter 35. Triton Inference Server Release 19.03

The TensorRT Inference Server container image, release 19.03, is available on NGC and is open source on GitHub.

## Contents of the Triton inference server container

The TensorRT Inference Server Docker image contains the inference server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04
- NVIDIA CUDA 10.1.105 including cuBLAS 10.1.105
- NVIDIA cuDNN 7.5.0
- NVIDIA NCCL 2.4.3 (optimized for NVLink™)
- OpenMPI 3.1.3
- TensorRT 5.1.2 RC

## Driver Requirements

Release 19.03 is based on CUDA 10.1, which requires NVIDIA Driver release 418.xx+. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384.111+ or 410. The CUDA driver's compatibility package only supports particular drivers. For a complete list of supported drivers, see the CUDA Application Compatibility topic. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.03 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.03 is based on NVIDIA TensorRT Inference Server 1.0.0, TensorFlow 1.13.1, and Caffe2 0.8.2.
- 19.03 is the first GA release of TensorRT Inference Server. See the README at the GitHub project for information on backwards-compatibility guarantees for this and future releases.
- Added support for "stateful" models and backends that require multiple inference requests be routed to the same model instance/batch slot. The new *sequence batcher* provides scheduling and batching capabilities for this class of models.
- Added GRPC streaming protocol support for inference requests.
- HTTP front-end is now asynchronous to enable lower-latency and higher-throughput handling of inference requests.
- Enhanced `perf_client` to support "stateful"/sequence models and backends.
- Latest version of NVIDIA CUDA 10.1.105 including cuBLAS 10.1.105
- Latest version of NVIDIA cuDNN 7.5.0
- Latest version of NVIDIA NCCL 2.4.3
- Latest version of TensorRT 5.1.2 RC
- Ubuntu 16.04 with February 2019 updates

## Known Issues

- If using or upgrading to a 3-part-version driver, for example, a driver that takes the format of `xxx.yy.zz`, you will receive a `Failed to detect NVIDIA driver version.` message. This is due to a known bug in the entry point script's parsing of 3-part driver versions. This message is non-fatal and can be ignored. This will be fixed in the 19.04 release.

# Chapter 36. Triton Inference Server Release 19.02 Beta

The TensorRT Inference Server container image, release 19.02, is available as a beta release and is open source on GitHub.

## Contents of the Triton inference server

This container image contains the TensorRT Inference Server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04
- NVIDIA CUDA 10.0.130 including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 10.0.130
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.4.2
- NVIDIA Collective Communications Library (NCCL) 2.3.7 (optimized for NVLink™ )
- OpenMPI 3.1.3
- TensorRT 5.0.2

## Driver Requirements

Release 19.02 is based on CUDA 10, which requires NVIDIA Driver release 410.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.02 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.02 is based on NVIDIA TensorRT Inference Server 0.11.0 beta, TensorFlow 1.13.0-rc0, and Caffe2 0.8.2.
- Variable-size input and output tensors are now supported.
- `STRING` datatype is now supported for input and output tensors for TensorFlow models and custom backends.
- The inference server can now be run on systems without GPUs or that do not have CUDA installed.
- Ubuntu 16.04 with January 2019 updates

## Known Issues

- This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.
- If using or upgrading to a 3-part-version driver, for example, a driver that takes the format of `xxx.yy.zz`, you will receive a `Failed to detect NVIDIA driver version.` message. This is due to a known bug in the entry point script's parsing of 3-part driver versions. This message is non-fatal and can be ignored. This will be fixed in the 19.04 release.

# Chapter 37. Triton Inference Server Release 19.01 Beta

The TensorRT Inference Server container image, release 19.01, is available as a beta release and is open source on GitHub.

## Contents of the Triton inference server

This container image contains the TensorRT Inference Server executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04
- NVIDIA CUDA 10.0.130 including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 10.0.130
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.4.2
- NCCL 2.3.7 (optimized for NVLink™ )
- OpenMPI 3.1.3
- TensorRT 5.0.2

## Driver Requirements

Release 19.01 is based on CUDA 10, which requires NVIDIA Driver release 410.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 19.01 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The inference server container image version 19.01 is based on NVIDIA TensorRT Inference Server 0.10.0 beta, TensorFlow 1.12.0, and Caffe2 0.8.2.
- Latest version of NVIDIA cuDNN 7.4.2
- Latest version of OpenMPI 3.1.3
- Custom backend support. The inference server allows individual models to be implemented with custom backends instead of by a deep learning framework. With a custom backend, a model can implement any logic desired, while still benefiting from the GPU support, concurrent execution, dynamic batching and other features provided by the inference server.
- Ubuntu 16.04 with December 2018 updates

## Known Issues

- This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.
- If using or upgrading to a 3-part-version driver, for example, a driver that takes the format of `xxx.yy.zz`, you will receive a `Failed to detect NVIDIA driver version.` message. This is due to a known bug in the entry point script's parsing of 3-part driver versions. This message is non-fatal and can be ignored. This will be fixed in the 19.04 release.

# Chapter 38. Triton Inference Server Release 18.12 Beta

The TensorRT Inference Server container image, previously referred to as Inference Server, release 18.12, is available as a beta release.

## Contents of the Triton inference server

This container image contains the TensorRT Inference Server (TRTIS) executable and related shared libraries in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04 including Python 3.5
- NVIDIA CUDA 10.0.130 including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 10.0.130
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.4.1
- NCCL 2.3.7 (optimized for NVLink™ )
- OpenMPI 3.1.2
- TensorRT 5.0.2

## Driver Requirements

Release 18.12 is based on CUDA 10, which requires NVIDIA Driver release 410.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384. For more information, see CUDA Compatibility and Upgrades.

## GPU Requirements

Release 18.12 supports CUDA compute capability 6.0 and higher. This corresponds to GPUs in the Pascal, Volta, and Turing families. Specifically, for a list of GPUs that this compute capability corresponds to, see CUDA GPUs. For additional support details, see Deep Learning Frameworks Support Matrix.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.12 is based on NVIDIA Inference Server 0.9.0 beta, TensorFlow 1.12.0, and Caffe2 0.8.2.
- TensorRT inference server is now open source. For more information, see GitHub.
- TRTIS now monitors the model repository for any change and dynamically reloads the model when necessary, without requiring a server restart. It is now possible to add and remove model versions, add/remove entire models, modify the model configuration, and modify the model labels while the server is running.
- Added a model priority parameter to the model configuration. Currently the model priority controls the CPU thread priority when executing the model and for TensorRT models also controls the CUDA stream priority.
- Fixed a bug in GRPC API: changed the model version parameter from string to int. **This is a non-backwards compatible change.**
- Added `--strict-model-config=false` option to allow some model configuration properties to be derived automatically. For some model types, this removes the need to specify the `config.pbtxt` file.
- Improved performance from an asynchronous GRPC frontend.
- Ubuntu 16.04 with November 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

# Chapter 39. Triton Inference Server Release 18.11 Beta

The inference server container image, previously referred to as Inference Server, release 18.11, is available as a beta release.

## Contents of the Triton inference server

This container image contains the Triton inference server executable in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04 including Python 3.5
- NVIDIA CUDA 10.0.130 including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 10.0.130
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.4.1
- NCCL 2.3.7 (optimized for NVLink™ )
- OpenMPI 3.1.2
- TensorRT 5.0.2

## Driver Requirements

Release 18.11 is based on CUDA 10, which requires NVIDIA Driver release 410.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384. For more information, see CUDA Compatibility and Upgrades.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.11 is based on NVIDIA Inference Server 0.8.0 beta, TensorFlow 1.12.0-rc2, and Caffe2 0.8.2.
- Models may now be added to and removed from the model repository without requiring an inference server restart.

- Fixed an issue with models that don't support batching. For models that don't support batching, set the model configuration to `max_batch_size = 0`.
- Added a metric to indicate GPU energy consumption.
- Latest version of NCCL 2.3.7.
- Latest version of NVIDIA cuDNN 7.4.1.
- Latest version of TensorRT 5.0.2
- Ubuntu 16.04 with October 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

# Chapter 40. Triton Inference Server Release 18.10 Beta

The Inference Server container image, previously referred to as Inference Server, release 18.10, is available as a beta release.

## Contents of the Triton inference server

This container image contains the Triton inference server executable in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04 including Python 3.5
- NVIDIA CUDA 10.0.130 including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 10.0.130
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.3.0
- NCCL 2.3.6 (optimized for NVLink™ )
- OpenMPI 3.1.2
- TensorRT 5.0.0 RC

## Driver Requirements

Release 18.10 is based on CUDA 10, which requires NVIDIA Driver release 410.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384. For more information, see CUDA Compatibility and Upgrades.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.10 is based on NVIDIA Inference Server 0.7.0 beta, TensorFlow 1.10.0, and Caffe2 0.8.2.
- Latest version of NCCL 2.3.6.
- Latest version of OpenMPI 3.1.2.

- Dynamic batching support is added for all model types. Dynamic batching can be enabled and configured on a per-model bases.
- An improved inference request scheduler provides better handling of inference requests.
- Added new metrics to indicate GPU power limit, GPU utilization, and model executions (which is useful for determining the impact of dynamic batching).
- Prometheus metrics are now tagged with GPU UUID, model name, and model version as appropriate, so that metric values can be correlated to specific GPUs and models.
- Request latencies reported by status API and metrics are more clear in what they report, for example total request time, queuing time, and inference compute time are now reported.
- Ubuntu 16.04 with September 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

# Chapter 41. Triton Inference Server Release 18.09 Beta

The Inference Server container image, previously referred to as Inference Server, release 18.09, is available as a beta release.

## Contents of the Triton inference server

This container image contains the Triton inference server executable in `/opt/tensorrtserver`.

The container also includes the following:

- Ubuntu 16.04 including Python 3.5
- NVIDIA CUDA 10.0.130 including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 10.0.130
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.3.0
- NCCL 2.3.4 (optimized for NVLink™ )
- OpenMPI 2.0
- TensorRT 5.0.0 RC

## Driver Requirements

Release 18.09 is based on CUDA 10, which requires NVIDIA Driver release 410.xx. However, if you are running on Tesla (Tesla V100, Tesla P4, Tesla P40, or Tesla P100), you may use NVIDIA driver release 384. For more information, see CUDA Compatibility and Upgrades.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.09 is based on NVIDIA Inference Server 0.6.0 beta, TensorFlow 1.10.0, and Caffe2 0.8.1.
- Latest version of cuDNN 7.3.0.

- Latest version of CUDA 10.0.130 which includes support for DGX-2, Turing, and Jetson Xavier.
- Latest version of cuBLAS 10.0.130.
- Latest version of NCCL 2.3.4.
- Latest version of TensorRT 5.0.0 RC.
- Google Cloud Storage paths are now allowed when specifying the location of the model store. For example, `--model-store=gs://<bucket>/<mode store path>`.
- Additional Prometheus metrics are exposed on the metrics endpoint: GPU power usage; GPU power limit; per-model request, queue and compute time.
- The C++ and Python client API now supports asynchronous requests.
- Ubuntu 16.04 with August 2018 updates

## Known Issues

- This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.
- Starting with the 18.09 release, the directory holding the Triton inference server components has changed from `/opt/inference_server` to `/opt/tensorrtserver` and the Triton inference server executable name has changed from `inference_server` to `trtserver`.

# Chapter 42. Inference Server Release 18.08 Beta

The NVIDIA container image of the Inference Server, release 18.08, is available as a beta release.

## Contents of the Inference Server

This container image contains the Inference Server executable in `/opt/inference_server`.

The container also includes the following:

- Ubuntu 16.04

Container image `18.08-py2` contains Python 2.7; `18.08-py3` contains Python 3.5.

- NVIDIA CUDA 9.0.176 (see Errata section and 2.1) including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 9.0.425
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.2.1
- NCCL 2.2.13 (optimized for NVLink™ )
- TensorRT 4.0.1

## Driver Requirements

Release 18.08 is based on CUDA 9, which requires NVIDIA Driver release 384.xx.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.08 is based on NVIDIA Inference Server 0.5.0 beta, TensorFlow 1.9.0, and Caffe2 0.8.1.
- Latest version of cuDNN 7.2.1.
- Added support for Kubernetes compatible ready and live endpoints.

- Added support for Prometheus metrics. Load metric is reported that can be used for Kubernetes-style auto-scaling.
- Enhance example `perf_client` application to generate latency vs. inferences/second results.
- Improve performance of TensorRT models by allowing multiple TensorRT model instances to execute simultaneously.
- Improve HTTP client performance by reusing connections for multiple inference requests.
- Ubuntu 16.04 with July 2018 updates

## Known Issues

- This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.
- There is a known performance regression in the inference benchmarks for ResNet-50. We haven't seen this regression in the inference benchmarks for VGG or training benchmarks for any network. The cause of the regression is still under investigation.

# Chapter 43. Inference Server Release 18.07 Beta

The NVIDIA container image of the Inference Server, release 18.07, is available as a beta release.

## Contents of the Inference Server

This container image contains the Inference Server executable in `/opt/inference_server`.

The container also includes the following:

- Ubuntu 16.04

Container image `18.07-py2` contains Python 2.7; `18.07-py3` contains Python 3.5.

- NVIDIA CUDA 9.0.176 (see Errata section and 2.1) including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 9.0.425
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.1.4
- NCCL 2.2.13 (optimized for NVLink™ )
- TensorRT 4.0.1

## Driver Requirements

Release 18.07 is based on CUDA 9, which requires NVIDIA Driver release 384.xx.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.07 is based on NVIDIA Inference Server 0.4.0 beta, TensorFlow 1.8.0, and Caffe2 0.8.1.
- Latest version of CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 9.0.425.
- Support added for TensorFlow SavedModel format.
- Support added for gRPC in addition to existing HTTP REST.

- Ubuntu 16.04 with June 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

# Chapter 44. Inference Server Release 18.06 Beta

The NVIDIA container image of the Inference Server, release 18.06, is available as a beta release.

## Contents of the Inference Server

This container image contains the Inference Server executable in `/opt/inference_server`.

The container also includes the following:

- Ubuntu 16.04

Container image `18.06-py2` contains Python 2.7; `18.06-py3` contains Python 3.5.

- NVIDIA CUDA 9.0.176 (see Errata section and 2.1) including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 9.0.333 (see section 2.3.1)
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.1.4
- NCCL 2.2.13 (optimized for NVLink™)
- TensorRT 4.0.1

## Driver Requirements

Release 18.06 is based on CUDA 9, which requires NVIDIA Driver release 384.xx.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.06 is based on NVIDIA Inference Server 0.3.0 beta, TensorFlow 1.8.0, and Caffe2 0.8.1.
- Support added for Caffe2 NetDef models.

- Support added for CPU-only servers in addition to servers that have one or more GPUs. The Inference Server can simultaneously use both CPUs and GPUs for inferencing.
- Logging format and control is unified across all inferencing backends: TensorFlow, TensorRT, and Caffe2.
- Gracefully exits upon receiving SIGTERM or SIGINT. Any in-flight inferences are allowed to complete before exiting, subject to a timeout.
- Server status is enhanced to report the readiness and availability of the server and of each model (and model version).
- Ubuntu 16.04 with May 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

# Chapter 45. Inference Server Release 18.05 Beta

The NVIDIA container image of the Inference Server, release 18.05, is available as a beta release.

## Contents of the Inference Server

This container image contains the Inference Server executable in `/opt/inference_server`.

The container also includes the following:

- Ubuntu 16.04

> Container image `18.05-py2` contains Python 2.7; `18.05-py3` contains Python 3.5.

- NVIDIA CUDA 9.0.176 (see Errata section and 2.1) including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 9.0.333 (see section 2.3.1)
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.1.2
- NCCL 2.1.15 (optimized for NVLink™)
- TensorRT 3.0.4

## Driver Requirements

Release 18.05 is based on CUDA 9, which requires NVIDIA Driver release 384.xx.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- The Inference Server container image version 18.05 is based on NVIDIA Inference Server 0.2.0 beta and TensorFlow 1.7.0.
- Multiple model support. The Inference Server can manage any number and mix of TensorFlow to TensorRT models (limited by system disk and memory resources).

- TensorFlow to TensorRT integrated model support. The Inference Server can manage TensorFlow models that have been optimized with TensorRT.
- Multi-GPU support. The Inference Server can distribute inferencing across all system GPUs. Systems with heterogeneous GPUs are supported.
- Multi-tenancy support. Multiple models (or multiple instances of the same model) can run simultaneously on the same GPU.
- Batching support
- Ubuntu 16.04 with April 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

# Chapter 46. Inference Server Release 18.04 Beta

The NVIDIA container image of the Inference Server, release 18.04, is available as a beta release.

## Contents of the Inference Server

This container image contains the Inference Server executable in `/opt/inference_server`.

The container also includes the following:

- Ubuntu 16.04 including Python 2.7 environment
- NVIDIA CUDA 9.0.176 (see Errata section and 2.1) including CUDA® Basic Linear Algebra Subroutines library™ (cuBLAS) 9.0.333 (see section 2.3.1)
- NVIDIA CUDA® Deep Neural Network library™ (cuDNN) 7.1.1
- NCCL 2.1.15 (optimized for NVLink™)

## Driver Requirements

Release 18.04 is based on CUDA 9, which requires NVIDIA Driver release 384.xx.

## Key Features and Enhancements

This Inference Server release includes the following key features and enhancements.

- This is the beta release of the Inference Server container.
- The Inference Server container image version 18.04 is based on NVIDIA Inference Server 0.1.0 beta.
- Multiple model support. The Inference Server can manage any number and mix of models (limited by system disk and memory resources). Supports TensorRT and TensorFlow GraphDef model formats.
- Multi-GPU support. The server can distribute inferencing across all system GPUs.
- Multi-tenancy support. Multiple models (or multiple instances of the same model) can run simultaneously on the same GPU.

- Batching support.
- Latest version of NCCL 2.1.15
- Ubuntu 16.04 with March 2018 updates

## Known Issues

This is a beta release of the Inference Server. All features are expected to be available, however, some aspects of functionality and performance will likely be limited compared to a non-beta release.

## Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

## VESA DisplayPort

DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources, and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Association in the United States and other countries.

## HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

## OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.



## Trademarks

NVIDIA, the NVIDIA logo, and cuBLAS, CUDA, CUDA Toolkit, cuDNN, DALI, DIGITS, DGX, DGX-1, DGX-2, DGX Station, DLProf, GPU, Jetson, Kepler, Maxwell, NCCL, Nsight Compute, Nsight Systems, NVCaffe, NVIDIA Ampere GPU architecture, NVIDIA Deep Learning SDK, NVIDIA Developer Program, NVIDIA GPU Cloud, NVLink, NVSHMEM, PerfWorks, Pascal, SDK Manager, Tegra, TensorRT, Tesla, TF-TRT, Triton Inference Server, Turing, and Volta are trademarks and/or registered trademarks of NVIDIA Corporation in the United States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

## Copyright