Release Notes

NiFi Registry 0.3.0

Version 0.3.0 of Apache NiFi Registry iterates on the capabilities introduced in previous releases that enable the storage/management of versioned NiFi flows.

Release Date: September 25, 2018

Highlights of the 0.3.0 release include:

- New Apache Ranger Authorizer allows authorization rules to be defined in Apache Ranger
- Expansion and refinement of the Versioned Flow data model to facilitate easier use by consumers
- Security enhancements: upgrades dependencies that have been patched for security issues
- Event Hook API improvements
    - New REGISTRY_START event type
    - Event Type Whitelisting feature for Event Hook Providers
- Docker image improvements

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?version=12343483&styleName=Html&projectId=12320920

Version 1.8.0

Version 1.8.0 of Apache NiFi is a feature and stability release.

Release Date: October 26, 2018

Highlights of the 1.8.0 release include:

- Secure cluster communication between nodes now requires two-way SSL
- Connections support load balancing across the cluster
- Nodes can be offloaded to prepare for decommissioning
- Easier to create a cluster on Docker with docker-compose and environment variables
- New or Improved Processors and Controller Services
    - LookupService that uses ElasticSearch
    - NetFlow Processors
    - JoltTransformRecord processor
    - Processors for interacting with Apache Kafka 2.0
    - SQL results can now be output as records in any supported format
    - ListenHTTP Processor supports multipart requests
- UI
    - Configure load balancing and load balancing compression from the Connection properties dialog
    - Connections indicate load balancing status
    - Cluster Summary dialog allows offloading disconnected nodes
- Documentation
    - New documentation for load balancing and node offloading
    - NiFi Port list

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12343482

Version 1.7.1

Version 1.7.1 of Apache NiFi addresses a few key bug fixes.

Release Date: July 16, 2018

Highlights of the 1.7.1 release include:

- Fixed an issue with wildcard certificates used to secure a cluster. Wildcard certificates are officially **not supported** but this regression behavior was fixed.
- Fixed an issue where an infinite loop could be caused by a stack overflow due to circular references.
- Fixed an issue with transitive controller service validation in unit testing.
- Fixed some issues related to processor termination and active thread representation.
- Fixed an issue with `X-Frame-Options` headers.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12343647

Version 1.7.0

Version 1.7.0 of Apache NiFi is a feature and stability release.

Release Date: June 25, 2018

Highlights of the 1.7.0 release include:

- New or Improved Processors, Controller Services, and Reporting Tasks
    - New XML record reader & writer.
    - New ForkRecord processor.
    - New SiteToSiteMetricsReportingTask.
    - Support for Hive 3.0.
- UI/UX
    - UI option to interrupt a running processor.
    - Improvements to cluster coordination.
- NiFi built against Java 1.8 can now run on Java 9.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?version=12342979&projectId=12316020

NiFi Registry 0.2.0

Version 0.2.0 of Apache NiFi Registry iterates on the capabilities introduced in the initial release that enable the storage/management of versioned NiFi flows.

Release Date: June 18, 2018

Highlights of the 0.2.0 release include:

- Adds a git flow persistence provider that stores flow snapshot versions in a git repository and can be optionally configured to push changes to a remote repository such as one hosted on GitHub.
- Adds a flow hook extension point API. Write and configure your own hooks that get called any time a new flow snapshot version is saved to NiFi Registry. An included implementation can be configured to run an external script any time a new flow version is created.
- Additional deployment flexibility and configuration options

    - An external database, such as PostgreSQL, can now be used in place of an embedded H2 database
    - Adds a Dockerfile for building a NiFi Registry Docker image
    - More options for mapping LDAP group membership of users to NiFi Registry when using LDAP security
- UI/UX refinements
    - Web UI now uses NiFi Flow Design System and Angular 5.2
    - Bucket IDs and Flow IDs are now discoverable in the UI to assist in the development of automations scripts
- Bug fixes
    - Flow version count updates correctly after adding a new version of a flow
    - Correct sorting of groups in the "Add user to groups" dialog
- REST API improvements
    - A new flow "diff" endpoint returns a summary of differences between two versions of a flow
    - New endpoints for interacting with flows by global id without having to specify the bucket
    - New remote management endpoints (specifically, Spring Boot Actuator endpoints are available)
    - A self-hosted Swagger UI to aid in development and testing
- Documentation improvements
    - A new "Getting Started" quick start guide has been added to the documentation
    - REST API documentation and Swagger specification corrections and additions. Documentation of each API endpoint now includes which access policies are required when NiFi Registry is secured

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12320920&version=12342369

NiFi Flow Design System 0.1.0

Version 0.1.0 of Apache NiFi Flow Design System is the initial release focusing on an atomic reusable platform for providing a common set of UI /UX components for Apache NiFi, Apache NiFi Registry, Apache NiFi MiNiFi, and any other open source web applications to consume.

Release Date: June 14th, 2018

Highlights of the 0.1.0 release include:

- Initial Implementation of the Flow Design Spec
    - Typography
    - Buttons
    - Inputs
    - Tabs
    - Tables
    - Checkboxes/Radios
    - Panels
    - Dropdowns
    - Modals
    - Notifications/Coasters
    - Accordians
    - Tooltips
- Themeable
- Unit tested components - 85% code coverage
- Sample application for demonstration, verification, and documentation of nifi-fds.

Version 1.6.0

Version 1.6.0 of Apache NiFi is a feature and stability release.

Release Date: April 8, 2018

Highlights of the 1.6.0 release include:

- New or Improved Processors, Controller Services, and Reporting Tasks
    - New REST endpoint exposed to gather processor diagnostics.
    - New processor to delete rows from Apache HBase.
    - New processor to put data to InfluxDB.
    - New processor and controller service to send data to Druid.
    - New processor to perform Mongo DB Aggregations.
    - New Processor to scan Apache HBase to retrieve records from tables.
- UI/UX
    - UI/Docs can now signal for key resource usage considerations based on annotated components flagging for things like high memory usage.
    - Users can now set fine grained policies for restricted components indicating the type of restriction such as
        - Reads Filesystem
        - Writes Filesystem
        - Executes Code
        - Keytab Access
    - Thanks to the new fine grained restricted component policies NiFi can now use its policy management mechanisms to protect specific keytab accesses in the flow making multi-tenant use cases easier and more secure.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12342422

Version 1.5.0

Version 1.5.0 of Apache NiFi is a feature and stability release focusing integration with the new Apache NiFi Registry.

Release Date: January 12, 2018

Highlights of the 1.5.0 release include:

- New or Improved Processors, Controller Services, and Reporting Tasks
    - Added Processors for interacting with Kafka 1.0
    - NiFi lineage can now be exported to Apache Atlas
    - HDFS, HBase, Hive processors updated to better handle Kerberos ticket renewal
    - Groovy Scripting can now leverage Grape/Grab through Ivy
    - Many processors and their properties now support expression language

- UI/UX
    - Users can now import, publish, change versions of flows via the NiFi and NiFi Registry Integration. This new feature takes care of starting and stopping affected processes, highlighting differences between local and published flows, and indicating when newer versions of a given flow are available.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12341668

NiFi Registry 0.1.0

Version 0.1.0 of Apache NiFi Registry is the initial release focusing on storage/management of versioned flows.

Release Date: January 1st, 2018

Highlights of the 0.1.0 release include:

- Initial REST API
    - Management of buckets, versioned flows, users/groups
    - Java client for interacting with the REST API
- Initial UI
    - Management of buckets and users via admin portion of the UI
    - Ability to view, sort, and filter versioned flows stored in buckets
    - Ability to view the version history for a given versioned flow
    - Ability to create and manage policies on buckets to restrict access to appropriate users
- Security
    - Authentication via certificates, LDAP, or Kerberos
    - Authorization via a file-based policy provider, with support for file-based and LDAP user-group providers

Version 1.4.0

Version 1.4.0 of Apache NiFi is a feature and stability release focusing on key bug fixes and new record based processors and controller services.

Release Date: October 2nd, 2017

Highlights of the 1.4.0 release include:

- Core Framework
    - OpenId Connect support for authenticating users
    - DistributedMapCacheClients and StateProviders can now use Redis
    - Can now use Apache Ranger to authorize groups and see Ranger policies in NiFi
    - Added support for LDAP-based user and group authorization
    - Added support for combining multiple user and group authorization providers in a chain
    - Added support for Knox (proxying and SSO)
    - NiFi now honors the "nifi.content.claim.max.appendable.size" property
    - Removed legacy dependency on Jasypt for string encryption; this is now implemented as native cryptographic components
- New or Improved Processors, Controller Services, and Reporting Tasks
    - Added Processors for interacting with Kafka 0.11.x supporting transactions and message headers
    - New processors for working with RethinkDB
    - New processors for sending data to Apache Kudu
    - gRPC client and server processors
    - New record-based processors: MergeRecord, ValidateRecord, ListenTCPRecord, PutHBaseRecord
    - In addition to the already supported simple Avro Schema Registry and Hortonworks Schema Registry, NiFi now supports integration with the Confluent Schema Registry
- UI/UX
    - Added support for Variable Registry at the Process Group level
    - Users can now double-click on a processor to display its configuration
    - When performing a Queue Listing, the UI will now indicate if the FlowFile has no content
    - Provenance repository storage usage is now displayed in the Systems Diagnostics dialog

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12340589

Version 1.3.0

Version 1.3.0 of Apache NiFi is a feature and stability release focusing on key bug fixes and new record based processors and controller services.

Release Date: June 8th, 2017

Highlights of the 1.3.0 release include:

- Clustering
  - Addressed issues with flow inheritability with regards to the scheduled state of each component
  - Improved stability when merging responses when lacking permissions to bulletins
  - Addressed issues when the active Cluster Coordinator is terminated without proper disconnection
  - Increased the default thread pool size for request replication
  - When joining a cluster, ensured components are not started until the local repositories have been initialized
  - Ensured response stream is accessible when attempting to view content
- Core Framework
  - Addressed timezone error when calculating component statistics
  - Added the ability to deprecate components
  - Addressed possible OutOfMemory exception when using an RPG with compression enabled
  - Addressed issue with swapping that could cause an NPE in the FileSystemRepository
  - Addressed issue with an RPG not honoring the yield duration when it was unable to contact the target instance
  - Ensured necessary response headers are set accordingly
- New or Improved Processors, Controller Services, and Reporting Tasks
  - Added UpdateRecord, PartitionRecord, LookupRecord, and PutElasticsearchHttpRecord processors
- UI/UX
  - Ensured that client revisions are correct when pointing to a node that becomes disconnected/connected to a cluster
  - Addressed issue that caused a JS bug when the component name was whitespace
  - Fix compatibility issues in deep linking with some older versions of FireFox
  - Addressed issue when attempting to change the version of the component that caused multiple versions to be returned
  - Addressed issue in the combo that caused the info icon to wrap (reported through long template names)
  - Performed additional validation of user supplied input

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12340498

Version 1.2.0

Version 1.2.0 of Apache NiFi is a feature and stability release that adds many new processors, record reader and writer services.

Release Date: May 8th, 2017

Highlights of the 1.2.0 release include:

- Core Framework
  - Now supports running multiple versions of the same components. Makes upgrades and multi-tenant flows easier to manage as sets the stage for upcoming Apache NiFi Registry work!
  - New provenance repository implementation available called 'WriteAheadProvenanceRepository'. Huge performance increase over the standard implementation and indexes in real-time.
- New or Improved Processors, Controller Services, and Reporting Tasks
  - New Record oriented abstraction for reading/writing schema aware event streams from CSV, JSON, AVRO, Grok, and plaintext with easy extension for other formats/schemas
    - QueryRecord processor to execute SQL queries over a stream of records powered by Apache Calcite
    - ConvertRecord processor to efficiently transform records from a given schema and format into another schema and format
    - SplitRecord processor to efficiently split huge record bundles into configurable batch sizes for divide and conquer or protect downstream systems
    - Processors to efficiently stream Records into and out of Apache Kafka in a format and schema aware manner and which automatically handle achieving high throughput and full provenance
    - Controller Services for plugging into and managing data schemas (Avro Schema Registry, Hortonworks Schema Registry) that integrate nicely into the record readers and writers
  - Features/improvements related to Change Data Capture (CDC), including CaptureChangeMySQL which reads from the MySQL binlogs, EnforceOrder, and PutDatabaseRecord processors, as well as a "Rollback on Failure" capability of some Put processors
  - New processors and controller service to support a Wait/Notify pattern enabling conditions in another portion of a flow to signal another portion to continue or execute.
  - For those that like to write new capabilities on the fly using scripting languages you can build your own reporting tasks, record readers, and writers using various scripting langauges now and the ExecuteScript processor lets you write in Clojure now too
  - The JSON Jolt Transform processor now allows Jolt transforms to include NiFi expression language statements and is much faster
  - New processors to compute and compare content using Fuzzy Hashing powerful for cyber security and other cases
  - New processors to interact with Google Cloud Platform/Google Cloud Storage and Azure Blob and Table Storage
  - Added ExtractCCDAAttributes processor to extract information from a Consolidated CDA formatted flow file
  - New reporting tasks available to push flow status and bulletins out via SiteToSite protocol
  - There are a 219 processors, 28 controller services, and 10 reporting tasks now available out the box. Check them out!
- User Interface/User Experience

  - Can't help but keep fiddling with processor alignment? Ok, a lot of us do that. You can now select numerous components and have it vertically or horizontally aligned!

- You can now deep link to components in the flow and share those links with your colleagues. Makes sharing and collaboration more natural
- Users will enjoy a better understanding of the scope of Controller Services through an improved experience
- For the web developer the UI is now modularized and circular references removed making it easier to extend and improve
- New content viewer available for AVRO data useful when clicking to content from provenance
- This release publishes the first official Apache NiFi Docker image to DockerHub!
- Provides a set of command-line utilities to aid in cluster management (add/remove nodes from cluster)
- Security
  - Users/client connecting to NiFi through the UI or API now protected with TLS v1.2 due to upgrade to Jetty version 9.4.2
  - The Apache Kafka processors for the 0.10 clients are on the latest and support dynamic JAAS configuration
  - The new provenance repository also has an implementation which supports full encryption for all events

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12338432

NiFi NAR Maven Plugin Version 1.2.0

Version 1.2.0 of the NiFi NAR Maven Plugin includes enhancements to support component versioning and addresses a bug fix.

Release Data: March 17, 2017

Highlights of the 1.2.0 release include:

- Added additional information to MANIFEST to support component versioning
- Added additional information to MANIFEST to allow a NAR to specify that it's resources should be cloned during instance class loading
- Fixed a bug when running the provided-nar-dependencies goal

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12339193

Version 1.1.2

Version 1.1.2 of Apache NiFi addresses a key bug fix.

Release Date: February 20, 2017

Highlights of the 1.1.2 release include:

- Minor user identity formatting refactoring to fix NullPointerException.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?version=12339600&projectId=12316020

Version 1.1.1

Version 1.1.1 of Apache NiFi addresses a few key bug fixes.

Release Date: December 22, 2016

Highlights of the 1.1.1 release include:

- Fixed an issue with a NullPointerException in Provenance Repo startup.
- Fixed an issue where the Provenance Repo wouldn't recover automatically when disk space runs out.
- Added missing images in the Admin Guide.
- Fixed scoping issues in templates related to Controller Services.
- Fixed policy issues relating to Remote Process Groups.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12338797

Version 1.1.0

Version 1.1.0 of Apache NiFi is a feature and stability focused release which builds on the great improvements and community progress of the 1.0 release.

Release Date: November 29, 2016

There are many changes in the 1.1.0 release with some highlights including:

- Core Framework Improvements
  - Cluster Management logic has been stabilized and improved to better support zero-master clustering. For example, recovery time is faster now as we're not having to always wait some artificial length of time to see if any new nodes will join and thus we should reconsider which flow is the golden copy.
  - The expression language capability now supports decimals, base64 and hex encoded values, and math functions.
  - The flowfile, provenance, and content repositories have been reworked to support rollback. We've always been careful to ensure that upgrades would work well where existing flow state would be honored but what was harder was supporting the ability to rollback where state was written via a new version of NiFi but the user decided to rollback to an older version. This is now supported. This powerful feature also sets the stage for future work to enable rolling upgrades and automated rollbacks!
  - Startup times for flows that have large backlogs should be far faster as the swap files have been reworked to provide summaries and avoid the need for full scans.
  - For developers, they can now indicate that their processor should be given an instance isolated classloader. Some libraries, like Hadoop client and scripting engines as an example, use static variables that can pollute instances of processors on the graph. This feature allows those cases to be easily overcome by enabling isolation per instance of a processor. This also makes it really easy to expose classloader extension for custom jars to users.
  - For developers, there is now an ability to migrate in-flight process session state to another process session. This will yield higher efficiency and makes for a far easier programming model for aggregation type patterns as seen in MergeContent, for example.
- User Experience Improvements
  - We now provide visual indication of queue growth relative to back pressure settings and when back pressure is engaged. This will make the concept of congestion and back pressure far more intuitive and frankly it is just fun to see in the UI. Definitely check this out.
  - After the 1.0.0 release several members of the community expressed how much they love the new look and feel but wished we had kept some of the colors. Better and more intuitive color contrast is back.
  - Validation of components is now limited to occur only for components which are not scheduled to execute. This results in much faster UX behavior as many of the operations one could do through the UI and REST API would result in expensive validation operations that were unnecessary.
  - Users can now export images of the provenance graphs.
  - Users can now use cron-scheduling for components even on primary node only tasks.
- Updated Versions of Dependencies
  - We now leverage the Azure Event Hubs 0.9.0 client library.
  - We now interact with Apache Spark using the 2.0.1 libraries.
  - We now interact with HDFS using the Apache Hadoop 2.7.3 libraries.
- New or Improved Processors
  - New Fetch and Put processors to interact with ElasticSearch 5.0 and new processors to execute Query and Scroll operations against ElasticSearch.
  - New processors to parse CEF formatted logs
  - The Extract Email processors now support TNEF formatted attachments.
  - New processor to validate CSV files.
  - The Apache Solr processors have been updated to support SSL and Kerberos.
  - New processors to act as client and server for Websockets.
- New Utility
  - In upgrading from 0.x to 1.x we provided a lot of capabilities to make the process easy and automatic. However, we didn't account for migrating from the embedded use of zookeeper to an external instance. We've now provided a utility that helps you migrate NiFi state from one zookeeper cluster to another.
  - Previously it was difficult to change the sensitive property key which is used to encrypt all sensitive properties contained within an actual flow configuration. A utility now exists to easily convert from an old key to a new key which is a valuable piece of an overall security process.
- Security Improvements
  - NiFi now supports the concept of restricted components. These are processors, controller services, reporting tasks that allow an authorized user to execute unsanitized code or access and alter files accessible by the NiFi user on the system NiFi is running. Therefore, these components are tagged by the developer as restricted and when running NiFi in secure mode only an administrator must grant each user access to the policy allowing restricted component access.
  - Site-to-Site now improved to be helpful even when port-forwarding is utilized. Very helpful for cases where an administrator might run NiFi with lower privileges but want external interaction to use well known privileged ports.
  - The policy management user experience has been improved to make it more intuitive what is happening in certain cases.
  - The encrypted configuration feature now has been extended to cover the Login Identity Provider capability. This is really helpful for example so you can have your LDAP password only stored in encrypted form in the login provider configuration file. Additional work is planned for these encrypted configurations to make interaction with a Hardware Security Module available as well.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12337875

Version 1.0.1

Version 1.0.1 of Apache NiFi backports a few key bug fixes.

Release Date: December 19th, 2016

Highlights of the 1.0.1 release include:

- Fixed multiple issues related to creating templates containing controller services.
- Fixed issue where PublishKafka could block indefinitely if fails to communicate properly to the broker.
- Fixed issue regarding the timing of Site-to-Site commits that could lead to data loss.
- Fixed permissions issues that prevented using the advanced tab of any processor in a secure instance.
- Fixed permissions issue that prevented downloading FlowFile content when using LDAP auth.

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12338865

Version 1.0.0

Version 1.0.0 of Apache NiFi is a major release that includes a brand new UI, framework level improvements and much more!

Release Date: August 30, 2016

There are many changes in the 1.0.0 release but the Application level highlights include:

- UI Refresh
    - The entire UI has been redone to have a cleaner, more modern look
    - UI has also been reworked to promote a better User Expierence
- Zero master clustering
    - There is no longer a NiFi Cluster Manager (NCM)
    - Cluster will now auto elect a Cluster Coordinator to oversee the cluster
- Multi-tenant authorization and internal authorization/policy management
    - There is no longer just the limited Administrator, Data Flow Manager, etc. roles
    - Policies are fully customizable for each component and at the controller level as well
- Deterministic template export
    - Templates export is now deterministic and thus can now be version controlled

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12332640

Version 0.7.4

Version 0.7.4 of Apache NiFi addresses several defects.

Release Date: June 8th, 2017

Highlights of the 0.7.4 release include:

- Fixes
    - Ensured necessary response headers are set accordingly
    - Performed additional validation of user supplied input

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12340590

Version 0.7.3

Version 0.7.3 of Apache NiFi addresses several defects and includes some performance and reliability improvements.

Release Date: May 17, 2017

Highlights of the 0.7.3 release include:

- Reliability
    - Resolved two issues that could cause provenance repository corruption
    - Resolved an issue that could cause files to not be deleted from the content repository
- Performance
    - Improved component validation to improve UI responsiveness when many components are on the graph
    - Improved documentation extration which reduces NiFi startup time
- Fixes

- Fixed defects in ControlRate, FetchSFTP, GetJMSQueue, GetJMSTopic, HandleHttpResponse, InvokeHTTP, PublishAMQP, PutJMS, PutKafka, and ReplaceText processors
- Fixed a defect in the nifi.sh start script
- Fixed a defect occuring when replaying files from a provenance display

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12340527

Version 0.7.2

Version 0.7.2 of Apache NiFi addresses two key bug fixes.

Release Date: February 20, 2017

Highlights of the 0.7.2 release include:

- Minor user identity formatting refactoring to fix NullPointerException.
- Regenerating test certificates for unit tests

A full list of issues that were resolved can be found at: https://issues.apache.org/jira/secure/ReleaseNote.jspa?version=12339601&projectId=12316020

Version 0.7.1

Version 0.7.1 of Apache NiFi focuses on addressing defects and security related improvements.

Release Date: October 20, 2016

Highlights of the 0.7.1 release include:

- Important stability improvements
    - Improved specific exception handling situations and synchronization in the **content repository** to avoid data loss.
    - Improved stability of **provenance repository** when handling exceptions during indexing of events.
- Improvements to existing capabilities
    - Added **SSL/TLS** support to **Kafka** processors.
    - Added features to **ConsumeKafka** to better align with features in GetKafka.
    - Improved **ConsumeJMS** to copy message properties into flowfile attributes, and to support more message acknowledgement modes.
    - Improved ConsumeJMS and PublishJMS for expression language support of destination name.
    - Various improvements to JoltTransformJSON, PutS3Object, ListS3, ListFile, ModifyBytes and PutEmail processors.
    - Deprecated some public API methods that were removed in the 1.0.0 release.
- Defects addressed
    - Fixed SiteToSiteReportingTask to not send duplicate events.
    - Fixed issue where controller services that reference other controller services could be disabled on NiFi restart.
    - Fixed issue with PublishKafka and PutKafka sending a flowfile to 'success' when it did not actually send the file to Kafka.
    - Fixed issue that multiple **HDFS processors** could not work with different HDFS clusters.
    - Fixed various issues with **Kafka processors**.
    - Fixed **GetHTTP** and **PostHTTP** to use the **SSL/TLS Protocol** property of its SSLContextService.
    - Fixed **UnpackContent** issues where it did not work with flowfile-stream or flowfile-tar content, and fixed concurrency issue.
    - Fixed ListenHTTP to avoid leaking threads when attempting to bind to an address already in use.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12338025

Version 0.7.0

Version 0.7.0 of Apache NiFi adds several valuable Processors, a new developer focused doc, and continued improvement to usability and stability.

Release Date: July 12, 2016

Highlights of the 0.7.0 release include:

- New Application Features
    - NiFi now supports exposing key/value pairs set in "nifi.properties" to **Expression Language.**
- Improvements to Existing Capabilities
    - Added the ability to bulk using Index or Update to **PutElasticSearch.**

- Allow concurrent execution of **ExecuteScript.**
- **Expression language** now has a random number function.
- New Extensions
  - New processors to Get data from **SNMP** agents and execute **SNMP** set requests.
  - Added new processor to Put to **Slack.**
  - New processors to Get and Put data to an **MQTT** broker.
  - Added processor to listen for messages using the **Lumberjack** protocol.
  - New **ExtractMediaMetadata** processor to extract metadata from various file types using Apache Tika.
  - New processors to Delete, Put and Get **AWS DynamoDB.**
  - Added processor to poll notification events provided by **HDFS Inotify** interface.
  - Add ReportingTask for sending **Provenance** events over **Site-To-Site.**
- Important Stability Improvements
  - Fixed instance where archiver was not respecting nifi.content.repository.archive.max.usage.percentage.
  - Resolved issue where incoming connection could be deleted while processor was still running.
  - Fixed issue where FileSystemRepository could leave open file handles to data that had already been archived and/or deleted.
- New Documentation
  - A new **NiFi in Depth** document was created to give an in-depth, developer level, look at NiFi and its design decisions.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?
projectId=12316020&version=12335078

Version 0.6.1

Version 0.6.1 of Apache NiFi focuses on addressing defects.

Release Date: April 18, 2016

Highlights of the 0.6.1 release include:

- Improvement to existing capabilities
  - Exposed property to control fetch size when interacting with databases in the 'QueryDatabaseTable' processor. Previously it just defaulted to one and would be innefficient.
- Defects addressed
  - Corrected an issue where high-rate bulletin generation caused more than the last five bulletins to be retained which wasted memory.
  - Corrected logic in GetKafka that resulted in leaking connections to Zookeeper ultimately leading to node instability.
  - Corrected PutKafka logic where incorrect message demarcation could result in data loss during transfer.
  - Simplified validation logic for Hadoop and HBase processors that led to excessive and expensive validation calls that caused sluggish UI behavior.
  - Resolved an edge condition which could cause the content archival to not expire data properly resulting in full content repositories and stopping the flow.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?
projectId=12316020&version=12335496

Version 0.6.0

Version 0.6.0 of Apache NiFi provides several new features and addresses multiple bugs or improvements.

Release Date: March 26, 2016

Highlights of the 0.6.0 release include:

- New Application Features
  - NiFi's REST API can now support **Kerberos Authentication** while running in an Oracle JVM.
- Improvements to Existing Capabilities
  - **ListenRELP** and ListenSyslog now alert when the internal queue is full. This means data receipt exceeds consumption rates as configured and data loss might occur so it is good to alert the user.
  - **ListHDFS** will now retain simple and finite state rather than an enumeration of all files seen with the same timestamp.
  - **ListenSyslog**, **ListenRELP**, **ListenUDP** have vastly improved performance and simpler configuration.
  - Updated Spark Receiver to support **Apache Spark** 1.6
  - Interaction with HBase now supports Kerberos enabled **Apache HBase**.
  - Fixed an issue in ExecuteSQL that caused SchemaParse failures against multiple databases.
- New Extensions
  - New processor '**QueryDatabaseTable**' monitors and tracks timestamp of latest record retrieved to support simple change capture cases.
  - Added a new processor that can be bootstrapped by a Spring Application Context. Could enable someone to run **Apache Camel** routes or **Spring Integrations** flows in NiFi for example.

- Added new processor to Put data with **Amazon Kinesis Firehose**
- New processor to put data to **Amazon Web Services Lambda** service.
- New processors to Put data to and Get data from **Splunk**.
- New processors to Get and Put data to **Apache Cassandra**
- Important Stability Improvements
  - Resolved corner case that could lead to corruption of flow file repository during shutdown.
  - Resolved a case that lead NiFi to hold open too many files when using run durations longer than 0 ms.
  - Resolved flow.xml.gz ordering issue that led to upgrade instability for clusters.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12334372

Known Issues with this release:

- **NIFI-1701** - Getting issue details...   STATUS   :PutKafka error handling and logic far better but poor stream demarcation can lead to data loss.

Version 0.5.1

Version 0.5.1 of Apache NiFi addresses several bugs and issues.

Release Date: February 26, 2016

Highlights of the 0.5.1 release includes:

- **Highlights of Bugs addressed**
  - **Close a case that could lead to data loss**: In the event that all flow files assigned to a given resource claim were swapped out, and the archive had to reclaim old space or was not engaged, and NiFi was restarted resource claims could be removed resulting in data loss.
  - **LDAP based authorization blocking valid API calls**: When using LDAP based authentication there were times when the access token wasn't being propagated which resulted in some valid API calls being rejected such as when downloading content or templates.
  - **Improved error messaging**: Several processors include improved error messages
  - **Hadoop processor bug fixed**: Issues with multiple PutHDFS processors fixed and client library dependency problems
  - **Clustering restart and upgrade reliability**: A long-standing issue which caused problems when a cluster was restarted or upgraded was fixed.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12334887

Version 0.5.0

Apache NiFi 0.5.0 includes several exciting new capabilities to include new processors, exciting new developer tools for building and testing processors. Stability and performance continue to be a priority and a long list of bugs were identified and resolved!

Release Date: February 16, 2016

Highlights of the 0.5.0 release includes:

- **New Application Features**
  - **Data inspection**: We've greatly enhanced the ability to interact with data and inspect it as it is flowing through NiFi.
- **Things to make NiFi Development Better and Easier**
  - **State Management**: As many developers are aware, keeping state in a Processor was often "up to you". This extension to the framework addresses this by adding state management as a core feature. In addition, many existing processors which kept state were modified to take advantage of this new capability.
  - **Testing Improvements**: We've improved the performance of many unit tests, added support for groovy for unit tests, the ability to test in unit tests, as well as better support for integration testing
- **Improvements to Existing Capabilities**
  - **RELP Support for Syslog**: In addition to adding support for RELP (Reliable Event Logging Protocol) for transporting syslog messages, emphasis was put on improving the framework to enable even more extensions in future releases.
  - **S3 Improvements**: We've pushed past the 5G max upload limitation of the previous PutS3Object processor by adding Multipart Upload support. Additionally, we added a couple authorization capabilities, like expression language support for keys, and an exciting new controller service created initially for fetching from buckets access across accounts.
  - **Hive Support**: The Kite bundle now supports sending data directly into Hive tables
  - **Encryption**: A broad set of enhancements were made to improve encryption and decryption of content and associated user documentation.
- **New Extensions**

- **Script Execution**: We've dramatically enhanced the ability to operate on data flowing through NiFi by adding support for launching scripts in a flow. We added a broad set of scripting languages - you can launch JRuby, Groovy, JavaScript, Lua, or Jython scripts with access to FlowFile data, which will allow much more dynamism and quick reaction capability in your data flows.
- **Support for sending Riemann Events**: NiFi now supports sending events to the Riemann event stream processing system.
- **Elasticsearch**: Connectivity to the popular search engine was added to this release
- **Avro Schema Inference**: Support was added allowing the inference of an Avro schema from JSON and CSV data flowing through NiFi.
- **AMQP support**: NiFi now supports sending and receiving from AMQP (Advanced Message Queueing Protocol) based messaging systems

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12334158

Known Issues with this release:

- **NIFI-1497** - Getting issue details... STATUS : An issue with Access Tokens may cause Viewing Content and Custom UIs to fail
- **NIFI-1527** - Getting issue details... STATUS : An issue that under certain conditions can result in data loss
- **NIFI-1694** - Getting issue details... STATUS : Users upgrading from NiFi 0.4.1 may encounter errors in PGP key-based encryption/decryption with `EncryptContent` processors if they have provided an individual key file for the `public keyring file` or `secret keyring file` property. A valid keyring file must be provided, and the full userID "`Name (Comment) <Email>`" form should be provided.

Version 0.4.1

Version 0.4.1 of Apache NiFi is an incremental release addressing several bugs and providing a few minor improvements over the 0.4.0 release.

Release Date: December 22, 2015

Highlights of 0.4.1 release include:

- **Bugs addressed**
    - **Site-To-Site**: If the remote system was applying back-pressure and the sending system attempted to stop the connection the sending system flow configuration could hang. If one of the nodes in a cluster being delivered to went off-line it could cause site-to-site to stop delivering to the other nodes in certain conditions. The automatic account request mechanism was broken.
    - **Flow File Ordering**: Ordering based on timestamp of flow file entry has been augmented with a sequence counter to provide better ordering precision.
    - **Run duration:** Only shown now when the processor supports the batching mechanism of run duration.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12334375

Version 0.4.0

Version 0.4.0 of Apache NiFi is a provides substantial improvements in functionality and usability, as well as providing some stability and performance improvements and bug fixes.

Release Date: December 11, 2015

Highlights of 0.4.0 release include:

- **New Application Features**
    - **Multiple Authentication Mechanisms**: NiFi now supports multiple Authentication Mechanisms! No longer is NiFi tied to being either non-secure or security based on two-way SSL but now can provide
    User Authentication via LDAP. This was a significant undertaking, but has paved the way to far more easily provide new Authentication Mechanisms. Future releases
    will include additional mechanisms, such as Kerberos.
    - **Drop FlowFiles from Queue**: Users are now able to right-click on a Connection and drop the FlowFiles in the queue, rather than relying on FlowFile Expiration to remove unwanted FlowFiles.
- **Usability Improvements**
    - **Explicit Processor Connectivity**: Processors that do not expect incoming data will no longer allow incoming Connections. Attempting to draw a connection to a "Source Processor" will show the
    connection line as a red, dotted line, and will not allow the Connection to be made. Likewise, Processors that require input to

perform work will be invalid until they have an incoming Connection.

Some Processors may accept incoming data or run without any incoming data. These Processors will be valid regardless of whether or not they have incoming connections.

- **Getting Started Guide**: Getting Started Guide is added to the 'help' screen of the application. This guide provides an introduction to NiFi terms, introduces the key concepts of NiFi and

  discusses how to work with FlowFiles and their Attributes. This is similar in concept to the User Guide but is far less verbose and explains concepts at a higher level.
- **Provenance Fetch Event**: A new Provenance Event Type (FETCH) was added. This Event Type is used to indicate that an existing FlowFile's contents were modified as a result of obtaining

  data from an external resource. This is in contrast to a RECEIVE event, which is used to indicate that a FlowFile entered the system as a result of obtaining data from an external resource.

- **New Extensions**
  - **RouteText**: Allows user to easily establish queries against textual data. Each line of a FlowFile is matched against the specified rules and routed according to the rules (potentially many lines of text

    are included in each output FlowFile). Also supports grouping of textual data so that FlowFiles that are output do not contain text from two different groups.
  - **TailFile**: Allows user to "tail" a file, consuming data from the end of the file as it is written by another process. This is typically used to consume data from log files as it is written. Processor will pick up

    where it left off, even if NiFi is restarted and log files roll over.
  - **ListenSyslog** / **PutSyslog**: Listens for incoming Syslog events over UDP or TCP and sends events to Syslog.
  - **ListFile** / **FetchFile**: Performs a listing of files in a given directory and fetches those files. These processors differ from GetFile in that the ListFile processor keeps state about files that have been consumed,

    so that the file can be ingested only once without deleting the source file. Additionally, if the directory being monitored exists in a mounted volume, the state can be shared across the cluster so that a new

    primary node can pick up where the previous processor left off, and the listing can also be shared across the cluster, distributing the work of pulling in and processing the files.
  - **ListSFTP** / **FetchSFTP**: Performs a listing of files on an SFTP server and fetches those files. Similarly to ListFile / FetchFile, state can be distributed across the cluster so that multiple nodes can perform the

    work in parallel.
  - **DeleteS3Object**: Removes an Object from Amazon S3.
  - **PutHBaseCell** / **GetHBase** / **PutHBaseJSON**: Allows users to put the contents of a FlowFile to HBase and listen for changes to an HBase table, automatically pulling in the rows that are added/updated.
  - **GetAzureEventHub** / **PutAzureEventHub**: Send the contents of FlowFiles as Events to Microsoft Azure Event Hub or listen for incoming Events on an Event Hub and create FlowFiles for those Events.
  - **GetCouchbaseKey** / **PutCouchbaseKey**: Send the contents of FlowFiles to Couchbase or fetch the contents of a record from Couchbase.
  - **AttributesToJSON**: Easily form a JSON document (as the contents of a FlowFile or as an Attribute) from a user-defined set of FlowFile Attributes.
  - **SplitAvro**: Splits a FlowFile that consists of many Avro records into individual FlowFiles, each containing a smaller number of Avro records.
  - **ExtractAvroMetadata**: Extracts the metadata from the header of an Avro file and adds the metadata to the FlowFile as a set of Attributes.
  - **Image Content Viewer**: When users look at the details of a Provenance Event, in the Content tab, if the View button is clicked, and the contents of the FlowFile are an image, that image will be rendered in

    the UI, rather than indicating that no viewer is available for this content type.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12333070

Version 0.3.0

Release Date: September 18, 2015

Highlights of 0.3.0 release include

- Performance improvements in handling large volumes of small files.
- Performance improvements in Provenance repositories.
- Added Reporting Task for Apache^{TM} Ambari^{TM}.
- Improved stability of nifi bootstrap.
- Added Processors for working with images.
- Support for interacting with Kerberos enabled Hadoop clusters
- Added additional Avro capabilities - merging datafiles & converting to json
- Added Processors for performing INSERT, UPDATE, DELETE statements against relational databases
- Added Processors that run Apache^{TM} Flume^{TM} sources and sinks.
- Added Processors to integrate with Amazon Web Services (S3, SNS, and SQS).
- Archival of content is now enabled by default.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12329653

NiFi NAR Maven Plugin Version 1.1.0

Release Date: August 23, 2015

Highlights of 1.10 release include

- Created a new plugin that allows developers to obtain a listing of transitive dependencies that are provided through a NAR dependency.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12333301

Version 0.2.1

Release Date: July 26, 2015

Highlights of 0.2.1 release include

- Removed all appropriate references to Incubator as NiFi is now a TLP.
- Fixed a bug that appeared in 0.2.0 that caused new users to a NiFi instance not to be prompted to create an account.

A full list of issues that were resolved can be found at https://issues.apache.org/jira/secure/ReleaseNote.jspa?projectId=12316020&version=12333089