Release Notes

In-Memory Data Fabric 6.6.5

- In-Memory HPC, Data Grid, and Streaming
  - Added missing ASYNC methods to GridCache
  - Added possibility to configure port for time server
  - Added possibility to configure internal thread pools
  - Fixed compute continuations to automatically cancel scheduled job restart
  - Fixed stability issue with cache file local store
  - Other fixes and improvements

In-Memory Data Fabric 6.6.4

- In-Memory HPC, Data Grid, and Streaming
  - Added event notification in case cache data is lost due to node(s) failure
  - Fixed stability issue with explicit (non-transactional) distributed locks
  - Fixed stability issue with distributed cache transactions
  - Fixed cache entries expiration with TTL set
  - Fixed an issue in GridOptimizedMarshaller with JDK8
  - Other fixes and improvements

In-Memory Data Fabric 6.6.3

- In-Memory HPC, Data Grid, and Streaming
    - Java and .NET interoperability:
        - Performance fixes for interop marshalling
        - Other fixes and improvements
    - Fixed issue with missing notifications for continuous queries with replicated cache
    - Other fixes and improvements

In-Memory Data Fabric 6.6.2

- In-Memory HPC, Data Grid, and Streaming
    - Java and .NET interoperability:
        - Added .NET data loader API
        - Added .NET continuous queries
        - Added ability to start GridGain as windows service
        - Added .NET grid lifecycle beans
    - Fixed issue with cache updates propagation while explicit lock is held on entry
    - Fixed several issues with conflict resolving for cache file local store
    - Fixed MXBeans registration for application servers clusters
    - Improved performance for offheap queries
    - Moved scalar examples to separate profile
    - Other fixes and improvements
- Visor Management and Monitoring
    - Performance fixes
    - Other fixes and improvements

In-Memory Data Fabric 6.5.6

- In-Memory HPC, Data Grid, and Streaming
    - Java and .NET interoperability: fixes and improvements
    - Portable object builder supports cyclic references
    - Portable object builder supports editing of nested objects without allocations overhead
    - Performance fixes for cache queries
    - Other fixes and improvements
- Visor Management and Monitoring
    - Added support for background execution of long-running tasks (e.g. SQL queries)
    - Improved SQL query editor
    - Improved new license loading: added license compare
    - Performance fixes
    - Other fixes and improvements

In-Memory Hadoop Accelerator 6.5.6

- Overall stability improvements with Hive
- Implemented performance counters with pluggable dumping facility
- Other fixes and improvements

In-Memory Data Fabric 6.5.5

- In-Memory HPC, Data Grid, and Streaming
    - Java and .NET interoperability: fixes and improvements
    - Portable Objects for Java - .NET - C++ interoperability
        - Improved object builder including ability to merge from multiple objects
        - Added support for `readResolve()` and `writeReplace()` methods
    - Added proxying for remotely deployed Distributed Services
    - Performance fixes for cache queries
    - Other fixes and improvements
- Visor Management and Monitoring
    - Performance fixes
    - Other fixes and improvements

In-Memory Data Fabric 6.5.2

- In-Memory HPC, Data Grid, and Streaming
    - Minor fixes and improvements
- Visor Management and Monitoring
    - Fixed maven artifacts deployment
    - Minor fixes and improvements

In-Memory Data Fabric 6.5.1

- In-Memory HPC, Data Grid, and Streaming
    - Added Off-Heap and Swap spaces to the scan queries
    - .NET Compute Grid
        - Full API parity with Java
        - .NET closure and compute task execution
- Visor Management and Monitoring
    - Removed dependency on Hadoop libraries from GGFS screen
    - Added two new series to Data Grid chart: "Number of swap keys" and "Number of Off-Heap Keys"

In-Memory Data Fabric 6.5.0

- In-Memory HPC, Data Grid, and Streaming
    - Combined all editions into one `in-memory data fabric` distribution
    - Java and .NET API parity
        - .NET Portable Cross Platform Objects
            - Portable Type Metadata
        - .NET Data Caching Node
        - .NET Client Node
        - .NET Transactions
            - Optimistic and Pessimistic concurrency
            - Read-committed, repeatable-read, and serializable isolation levels
        - .NET Distributed Locks
        - .NET Near Cache
        - .NET Cache Affinity
        - .NET Grid and Cache Projections
        - .NET Cluster Leader Election
    - Removed dependency between GGFS (GridGain File System) and Hadoop libraries
    - Added support for Spring 4.x
    - Added automatic warmup capability
    - Fixed slow downs on topology node crashes
- Visor Management and Monitoring
    - Added Portable Object Metadata screen for viewing portable types structure

In-Memory Hadoop Accelerator 6.5.0

- Fixed class loading issues with running multiple jobs
- Fixed memory utilization issues
- Other stability and performance fixes

In-Memory Computing Platform 6.2.1

- Security and Audit
    - Added task name or task class name for all cache `READ` events if they happen from within task execution
    - Added `READ` event in addition to `PUT` event for transform operations
    - Added transformer class name to all cache `READ` and `PUT` events if they happen within a transformer
    - Ensured that proper `READ` event is triggered for all query executions, i.e. SCAN, SQL, TEXT, and Continuous queries
- Added Spring `@Cacheable` annotation support
- Fixed license processor to calculate CPUs, hosts, etc. only for nodes with same cache distribution mode
- Fixed DR to work with portable configuration
- Additional fixes and improvements
- Visor Management and Monitoring
    - Added quick filters by message type for event logs

In-Memory Computing Platform 6.2.0

- In-Memory HPC, Data Grid, and Streaming

        - Added Portable Objects for Java - .NET - C++ interoperability.
            - Support for structural data changes without cluster restart.
            - Support for basic user object serialization across platforms (able to treat any object as portable object).
            - Automatic handling for Maps and Collections across platforms.
            - Ability to specify custom serialization logic through `GridPortableMarshalAware` interface.
            - Automatic handling of primitive types, dates, timestamps, and UUIDs across platforms.
            - Added portable type and field metadata.
            - Support for custom serializers to provide custom serialization logic without changes to portable classes.
            - Support for customer type ID and field ID mappers.
            - Ability to specify portable objects from client side with star "*" wildcard notation (e.g. "my.portables.package.*").
            - Ability to specify alternate affinity key for portable objects.
        - Added Distributed Services to support cluster-aware service deployments.
            - Support for per-cluster singleton services.
            - Support for per-node singleton services.
            - Support for service deployment with configurable number of instances per node or per cluster.
            - Support for service deployment based on cache affinity key.
            - Support for automatic service deployment on startup or dynamically through API.
        - Support for automatic leader election with new `forOldest()` and `forYoungest()` methods added to `GridProjection`.
        - Ability to read license file from class path using `classpath:some/path` syntax.
        - Made JDBC table creation optional in `GridCacheJdbcBlobStore` and `GridTcpDiscoveryJdbcIpFinder`.
    - Visor Management and Monitoring
        - Added limited Visor edition for open source users.
        - Added new tab for rolling updates.
        - Added new security tab with a list of authenticated subjects and permissions.
        - Added Visor authentication for cases when security is enabled.
        - Added support for custom (pluggable) tabs in Visor.
        - Added support for executing alert scripts in addition to sending alert emails.
        - Added "Node busy time percentage" chart series.
        - Added support for "*" and "?" wildcard symbols to all table filters.
        - Added Visor connectivity through SSH tunnel.
    - In-Memory Hadoop Accelerator
        - Added support for Apache Hadoop 2 compatible In-Memory Map-Reduce.
        - Changed GGFS URI structure.
        - Implemented in-process Hadoop file system wrapper for GGFS.
        - Removed support for Apache Hadoop 1.
        - Implemented command line setup tool for Apache Hadoop client configuration.

In-Memory Computing Platform 6.1.9

- Fixed problems with `gridgain-jta.jar`
- Visor management and monitoring
    - Fixed dependency list

In-Memory Computing Platform 6.1.8

- Project modularization and dependency cleanup
- Other bug fixes and enhancements
- Not backward compatible with GridGain Enterprise 6.1.5 and 6.1.6 (all releases before and after are OK).
- Visor management and monitoring
    - File Manager: added ability to search in subfolders
    - Updated scala libs from 2.10.3 to 2.10.4 version
    - Other bug fixes and enhancements

In-Memory Computing Platform 6.1.7

- Added client authenticated subject cache shared across nodes.
- Added authenticated subject ID to Cache and Task events.

In-Memory Computing Platform 6.1.6

- Fixed MVCC locking for REPLICATED cache
- Added socket buffer size adjustment for better performance in TCP communication
- Fixed NPE for NEAR_ONLY caches when deployed with PARTITIONED_ONLY servers

In-Memory Computing Platform 6.1.5

- Added java.util.Set implementation to distributed cache datastructures
- Added support for synchronous evicts in replicated cache
- Added GridCacheInterceptor for intercepting and modifying cache operations
- Added GridSecurity facade
- Added authorization for main cache and task operations
- Significant performance improvements to remote clients
- Other bug fixes and enhancements
- Visor management and monitoring
    - Added support for search and replace in configuration and Sql editors.
    - Sql Viewer Tab: added possibility to change query "Page Size" (number of rows to fetch at once) at any moment
    - Extended fiters for charts on Datagrid Tab (added filtering by caches), GGFS Tab (added filtering by nodes), Streamimg Tab (added filtering by streamers)
    - Other bug fixes and enhancements

In-Memory Computing Platform 6.1.1

- Enhancements for JDK8 lambdas support
- Enhancements and bug fixes for rendezvous affinity function

In-Memory Computing Platform 6.1.0

- Added support for distributed execution of JDK8 lambdas
- Added support for rendezvous affinity function and "fair" affinity function for better partition distribution between data nodes
- Added support for geospatial indexes in data grid queries
- Significant performance improvements in distributed queue implementation
- Fixed problem with multicast IP finder in some environments
- Enhanced events API
- Other bug fixes and enhancements
- Visor management and monitoring
    - Enhanced data center replication UI
    - Other bug fixes and enhancements

In-Memory Computing Platform 6.0.3

- Added support for atomic mode for distributed cache with near cache enabled
- Added support for atomic mode for local cache
- Datacenter replication: improvements and bug fixes and documentation improvements
- REST configuration made optional
- Other bug fixes and enhancements
- Visor Management and monitoring
    - Fixed issues when more than 1 Visor are connected to the grid
    - Other bug fixes and enhancements

In-Memory Computing Platform 6.0.2

- Simplified configuration
- Simplified documentation
- Migrated to Maven Central
- Performance improvements for continuous queries
- Other bug fixes and enhancements
- Visor Management & Monitoring
    - Improvements to datacenter replication tab
    - Added ability to view existing indexes
    - Added ability to scan cache contents even with indexes disabled (supported in command line visor as well)

In-Memory Computing Platform 6.0.1

- Removed all deprecated code
- Added Bi-directional WAN Datacenter Replication across different geographies
- Added local restartable store for disk-based recovery in case of large cluster failures
- Added ATOMIC mode for REPLICATED cache
- Added Near cache and Client-only mode to REPLICATED cache
- Added Hibernate L2 cache out-of-the-box integration
- Added GridLifecycleAware callbacks to all configurable grid components

- Significant performance improvements to network I/O
- Significant performance improvements to C++ client
- Significantly simplified all examples and migrated them to Maven for easier integration with IDEs
- API changes (not backward compatible)
    - Refactored GridProjection into following components and significantly simplified APIs
        - GridProjection - clustering and logical grid node grouping
        - GridCompute - distributed computations on the grid
        - GridMessaging - topic-based message exchange functionality
        - GridEvents - distributed event notifications
        - GridScheduler - cron-based scheduling functionality
        - GridProduct - product information and licensing functionality
    - Refactored GridCacheProjection into following components and significantly simplified APIs
        - GridCacheProjection - data manipulation, transactions, and locking functionality
        - GridCacheAffinity - data partitioning and key-to-node mapping functionality
        - GridCacheQueries - SQL, scan, and full text search queries
        - GridCacheDataStructures - various grid-aware data structures similar in APIs to `java.util.concurrent` package
    - Removed GridFunc and significantly simplified `org.gridgain.grid.lang` package
    - Removed `typedef` package and replaced all class aliases with class names
- Various other fixes and performance improvements
- Visor Management & Monitoring
    - New tabs for datacenter replication send and receive counterparts
    - Added SQL syntax highlighting to SQL Queries tab
    - Performance improvements to command-line Visor (moved it off of Scala REPL)

In-Memory Data Grid 5.3.3

- Fixed slow memory leak in query execution.
- Fixed slow memory leak in the lock methods in cache projection with predicate.

In-Memory HPC/Data Grid 5.3.1, Streaming 2.1.1, Hadoop Accelerator 2.1.1

- Rolling updates feature in this release is not compatible with previous releases due to significant enhancements to messaging protocol.
- Added auto-unsubscribe of continuous queries and continuous event notifications if requester node leaves grid.
- Added binding to all network interfaces on startup if local host is not explicitly specified.
- Visor Management New Features and Enhancements
    - Added ability to open multiple node tabs at once.
    - Added ability to view files directly from GGFS File Manger either using system viewer or internal visor viewer.
    - Added execution plan option to SQL Viewer tab.

In-Memory HPC/Data Grid 5.3.0

- Fixed issues related to memory consumption.
- Added eager cache entry expiration based on time-to-live.
- Cleaned up examples structure.
- Enhancements and bug fixes.

Streaming 2.1.1, Hadoop Accelerator 2.1.1

- Enhancements and bug fixes.

In-Memory HPC/Data Grid 5.2.2, Streaming 2.0.2, Hadoop Accelerator 2.0.2

- Added support for Web Session caching, automatic web session fault tolerance, load balancing, and expiration.
- Added magic bytes to discovery messages to make sure that other applications would not be send messages to GridGain.
- Added automatic back-pressure for FULL_ASYNC write synchronization mode to avoid possible out-of-memory condition under load.
- Added API and performance optimizations for synchronous query execution.
- Other bug fixes and enhancements.
- Visor Management New Features and Enhancements
    - Added Visor remote connectivity support. Now you can use Visor to monitor remote grids without having to become a grid member. For example, you can have your grid running on a public cloud and start Visor Console on your local laptop.

- Fixed issue when CPU was greater than 100% on charts.

In-Memory HPC/Data Grid 5.2.1, Streaming 2.0.1, Hadoop Accelerator 2.0.1

- Fixed backward version compatibility issue

In-Memory HPC/Data Grid 5.2.0 Release Notes

- Fixed slow memory leak in query execution.
- Fixed slow memory leak in the lock methods in cache projection with predicate.
- Added Multicast-based IP Finder to GridTcpDiscoverySpi.
- Added backward compatibility support to allow different versions of GridGain run in the same cluster.
- Added rich performance suggestions on startup.
- Implemented PRIMARY_SYNC mode for partitioned transactional cache.
- Significantly reduced size of cache entry.
- Changed node consistent hash ID from node ID to ip:port combination to ensure that node remains in same consistent hash position upon restart.
- Deprecated evictionEnabled flag.
- Implemented optional asynchronous message sending based on NIO functionality (in addition to existing NIO-based message receiving).
- Implemented clock-versioning in distributed caches to speed up FULL_SYNC mode.
- Removed serialization of values put in cache or off-heap when value is of primitive byte-array type.
- Significantly improved error messages and usability suggestions.
- Added performance suggestions for non-optimal configurations.
- Numerous enhancements and bug fixes.

In-Memory Streaming 2.0 Release Notes

- Added Hash-based and Tree-based indexes for rolling windows.
- Numerous enhancements and bug fixes.

In-Memory Accelerator for Hadoop® 2.0 Release Notes

- Added MapReduce over GGFS functionality.
- Numerous enhancements and bug fixes.

In-Memory HPC/Data Grid 5.1.6 Release Notes

- Added CLIENT_ONLY mode for partitioned cache.
- Added ATOMIC atomicity mode which provides better performance for non-transactional use cases.
- Added optional GridOptimizedMarshallable interface to help optimized marshaller remove internal lookups.
- Added one-phase commit in TRANSACTIONAL mode for basic put and putAll operations.
- Added automatic back-pressure control for async operations.
- Multiple fixes/enhancements to Visor Management Console

In-Memory Accelerator for MongoDB® 1.0b Release Notes

- Available in GridGain Maven Repository.
- Support for all MongoDB commands except for aggregation framework.
- Support for automatic repartitioning.
- Provided benchmarks to compare performance with native MongoDB.
- Visor Management New Features and Enhancements
    - New Visor MongoDB tab for management and monitoring of Mongo databases and collections stored in GridGain.

In-Memory Accelerator for Hadoop® 1.2 Release Notes

- Available in GridGain Maven Repository.
- Visor Management New Features and Enhancements

- Improved time accuracy in Profiler tab to nanoseconds.
- Default folders for every operational mode are optionally created upon GGFS (re)format.
- Ability to turn logging on and off when profiler starts and stops.

In-Memory Streaming 1.1 Release Notes

- Available in GridGain Maven Repository.
- Minor bug fixes.
- Visor Management New Features and Enhancements
    - Added query metrics to Streaming tab.

In-Memory HPC/Data Grid 5.1.2 Release Notes

- Available in GridGain Maven repository
- Added direct client-to-node connectivity mode for remote clients.
- Added type-based filtering to `Continuous Query` functionality.
- Added `AutoCloseable` to all main grid APIs.
- Added `GridCacheCountDownLatchExample`.
- Visor Management New Features and Enhancements
    - Fixed partition ID formatting in Database tab.

In-Memory Accelerator for Hadoop® 1.0 Release Notes

- Available in GridGain Maven repository.
- New Features and Enhancements
    - Added support for Hadoop 2.0 in addition to Hadoop 1.0.
    - Added DUAL_ASYNC mode to allow asynchronous writes to HDFS.
    - Added asynchronous delete in GGFS.
    - Added formatting function to GGFS.
    - Added additional metrics to GGFS.
- Visor Management New Features and Enhancements
    - Added hot-key support to File Manager tab.
    - Added enhanced search capabilities to File Manager tab.
    - Added Profiler tab to profile GGFS and HDFS performance and identify hotspots on the fly
    - Ability to profile various read and write metrics.
    - Access uniformity to control which sections of the file get accessed the most.
    - Ability to take profiling snapshots.
    - Ability to clear and reset profiler data.
    - Ability to filter based on GGFS mode as well as file names.

In-Memory Streaming 1.0b (Beta) Release Notes

- Available in GridGain Maven repository.
- New Features and Enhancements
    - Direct support for Complex Event Processing (CEP).
    - Pluggable rolling event windows
        - Support for batch windows.
        - Support for unique and sorted windows.
        - Support for bounded and unbounded windows.
    - Pluggable routing.
    - Affinity event collocation with GridGain in-memory data grid nodes.
    - Branching pipelines.
- Visor Management New Features and Enhancements
    - New Streaming tab to manage and monitor all deployed streamers including.

In-Memory HPC/Data Grid 5.1 Release Notes

- Available in GridGain Maven repository

- New Features And Enhancements
    - New continuous querying capability.
    - New continuous remote event notification capability.
    - Implemented efficient IPC communication between multiple nodes started on the same host.
    - Added off-heap metrics to GridCacheProjection.
    - Added examples for direct GGFS API usage.
    - Added EVT_CACHE_PRELOAD_OBJECT_LOADED and EVT_CACHE_PRELOAD_OBJECT_UNLOADED to notify listeners if an object got pre-loaded or unloaded by the system.
- Visor Management New Features and Enhancements
    - Many convenience and usability enhancements to UI.
- Core Bug Fixes
    - Deprecated GridJobAdapterEx in favor of GridJobAdapter and GridJobContinuationAdapter.
    - Fixed NotSerializableException in GridOptimizedMarshaller in JDK7.
    - Multiple bug fixes for IPC communication.

Gridgain 5.0 Release Notes

- New Features And Enhancements
    - New off-heap memory mode allowing to cache all keys on-heap and all values off-heap.
    - New `GridFileSwapSpaceSpi` which caches all keys in-memory and keeps all values on local file system.
    - GGFS - GridGain distributed in-memory file system.
    - `GridGgfsHadoopFileSystem` - Hadoop FileSystem adapter for GGFS to be used together with or instead of HDFS or any other Hadoop file system.
    - Support of automatic read-through and write-through to and from HDFS or any other Hadoop file system.
    - Support for high-throughput IPC over shared memory on Linux.
    - Series of GGFS vs. HDFS benchmarks to demonstrates performance differences.
    - Improved IO and thread context switching to achieve significant performance improvements for high-throughput messaging and job execution.
- Visor Management New Features and Enhancements
    - New GGFS tab to manage and monitor various GGFS properties.
    - New FileManager tab to provide various file system operation within the same file system and across different file systems.
- Core Bug Fixes
    - Fixed handling of spaces in Windows `.bat` scripts.

GridGain 4.5.0 Release Notes

- New Features And Enhancements
    - `HyperLocking` to minimize locking and serialization overhead for cache transactions under load.
    - `Risk Analytics` benchmark.
    - Custom SQL Functions including `GridCacheQueryCustomFunctionExample` to show how to use them.
    - Full off-heap indexing to `GridH2IndexingSpi`.
    - Execution plan printout for long queries by setting `GridH2IndexingSpi.setLongQueryExplain(true)` parameter.
    - Topic-based user message exchange.
    - `GridNoopCheckpointSpi` to remove checkpoint overhead whenever checkpoints are not used.
    - `GridNoopSwapSpaceSpi` to remove swap space overhead whenever it is not used.
- Visor Management New Features and Enhancements
    - Added `telemetry` screen in Visor to show overall grid status based on various metrics.
    - Added dedicated `cache` tab to show all cache-specific information.
- Core Bug Fixes
    - Path space issues in `ggstart.bat` startup script.
    - Deadlock with concurrent `evictAll()` and `unswapAll()`.
    - Query iterators are removed but not closed when originating node leaves of fails.
    - Restructured all examples to make them easier to use and understand.
- Client Connectivity Bug Fixes
    - Removed `ADD` method from client API as it was identical to `putIfAbsent` method.
- Visor Management Bug Fixes
    - Visor graph tooltip does not show whole information.
    - Visor spits errors (failed to fetch model update) when new node joins and busy with data pre-loading.

GridGain 4.3.1 Release Notes

- New Features and Enhancements
    - Added remove operation to data loader.
    - Significantly improved performance of partition to node mapping.
    - Added `GridSerializationBenchmark` for comparing performance of Java, Kryo, and GridGain serialization.
    - Added property-based configuration to remote clients.
    - Optimized concurrency for asynchronous methods in C++ client.
- Core Bug Fixes
    - Unmarshalling of `SimpleDateFormat` fails with NPE.
    - Possible NPE in Indexing Manager when using distributed data structures.
    - Swap partition iterator skips entries if off-heap iterator is empty.
    - `GridDataLoader` does not allow to cache primitive arrays.
    - Excessive memory consumption in indexing SPI.
    - Add check on startup that `GridOptimizedMarshaller` is supported by running JDK version.
    - If ordered message is timed out, other messages for the same topic may not be processed.
    - `ScalarPiCalculationExample` does not provide correct estimate for PI.
- Client Connectivity Bug Fixes
    - Client router with explicit default configuration leads to NPE.
    - Repair REST client support to make session token and client ID optional.
    - Ping does not work properly in C++ client.
- Visor Management Bug Fixes
    - Clear and Compact operations in Visor do not account for node selection.
    - Move Visor management tasks into a separate thread pool.
    - Preload dialog in Visor does not show correct number of keys.
    - GC dialog in Visor waits indefinitely for dead nodes.
    - Increase tooltip dismiss time in Visor.
    - Visor log search does not show nodes table correctly on Windows.

GridGain 4.x Release Notes

- In-Memory Data Grid / Distributed Cache.
- Off-Heap BigMemory Storage to avoid lengthy GC pauses when using large amounts of memory.
- Indexing SPI for pluggable highly concurrent in-memory indexing.
- Data Loader for efficient bulk-load operations on cache.
- Distributed data structures
    - Distributed Atomic Sequence.
    - Distributed Atomic Long.
    - Distributed Stamped Reference.
    - Distributed Count Down Latch.
    - Distributed bounded and unbounded queues.
- Near-only cache (enabling cache operations without participating in caching).
- Partition-only cache.
- Significant optimizations for Collocated cache operations.
- Delayed and manual repartitioning to avoid excessive network traffic on topology changes.
- Write-behind caching.
- Synchronized cache evictions with other cache nodes.
- Distributed cache garbage collector to monitor/remove stale locks.
- Significantly reduced cache memory consumption.
- Compute Grid
    - Significant optimizations on all levels utilizing concurrent data structures.
    - Distributed Continuations and Recursive Split.
- Remote Client Connectivity
    - Affinity-aware Java Remote Client.
    - Affinity-aware .NET Remote Client.
    - Affinity-aware C++ Remote Client.
    - REST-based HTTP Remote Client.
    - Affinity Aware Router/Gateway for remote client connectivity.
    - JAAS-based authentication of remote clients.

        - Secure communication protocol for establishing secure channels between remote clients and grid nodes
            - RememberMe secure session SPI.
        - Clustering / Networking
        - Pluggable Network Segmentation Policies to handle Split-Brain Network Segmentation
            - Basic reachability segmentation resolver.
            - Shared file system segmentation resolver.
            - TCP connection segmentation resolver.
        - Support for authentication of cluster nodes
            - JAAS-based authentication SPI.
            - Passcode-based authentication SPI.
- Marshalling
    - Optimized Marshalling achieving 20x faster serialization than standard Java serialization.
- Management And Monitoring.
- Visual GUI-based Visor Management And Monitoring
    - Dashboard Tab
        - View CPU and Heap Charts.
        - View grid topology and metrics for all grid nodes.
        - Start/stop/restart any number of remote nodes.
        - Dynamically view/update license at runtime.
        - Integration with VisualVM for automatic JVM profiling of remote nodes.
        - Visual thread dump of any grid node.
        - View and search logs from any of the nodes.
        - View Grid events.
    - Group Tab
        - Provides functionality identical to Dashboard for any subgroup of the nodes.
    - Host Tab
        - Provides functionality identical to Dashboard for all nodes started on a specific host.
    - Node Tab
        - Provides functionality identical to Dashboard for a specific node.
        - Various views of metrics, configuration, and system or environment properties for given node.
    - In-Memory HPC Tab
        - View various charts for currently executing and waiting jobs.
        - View tasks executions including split information and nodes assignments.
        - Ability to map tasks and jobs to nodes and vise versa.
    - In-Memory Data Grid Tab
        - View various charts regarding cache hits/misses/reads/writes/commits/rollbacks.
        - View statistical information about all caches defined.
        - Ability to map from caches to nodes and from nodes to caches.
        - Execute various management commands such as *compact, clear, swap, preload, load,* etc…
    - Object Browser Tab
        - View metadata/schema of all cached deployed in the grid.
        - Run SQL queries on any of the nodes and caches using built-in SQL viewer.

GridGain 3.x Release Notes

- In-Memory Data Grid
    - Distributed in-memory key-value store.
    - Local / Replicated / Partitioned caches.
    - Full support for ACID transactions
        - Optimistic / Pessimistic transactions.
        - Read-Committed / Repeatable-Read / Serializable isolation levels.
        - JTA / JCA Integration.
    - Disk-based Swap overflow storage
        - LevelDB swap storage SPI.
    - Pluggable eviction policies
        - LRU
        - FIFO
        - Random
        - Time-based
    - MVCC-based concurrency.

        - Synchronous and Asynchronous cache operations.
        - Replication and Invalidation modes.
        - Read-Through and Write-Through behavior for pluggable GridCacheStore.
        - Pluggable data partitioning.
        - SQL Queries for in-memory data
            - Affinity-based collocated queries.
            - Predicate-based full-scan queries.
            - Remote query transformations.
            - Local and Remote query result reduction.
            - Lucene-based text queries.
        - H2-based Data Indexing for SQL queries.
        - OOP and FP-based APIs for Java and Scala.
        - Automatic support for object cloning.
- Compute Grid
    - Distributed cache as checkpoint storage.
    - Affinity routing for partitioned in-memory data grid.
    - GridNodeLocal cache for storing shared state between jobs.
    - Distributed continuations for suspending/resuming job execution.
    - Cron-based scheduling.
- Clustering / Networking
    - TCP-based Discovery SPI.
        - Pluggable IP-Finders.
            - In-VM IP Finder
            - Amazon S3 IP Finder
            - Shared-FS IP Finder
        - Support for 1000+ node topologies.
- Management And Monitoring
    - Shell-based Visor Management And Monitoring.
        - Support for various grid topology and configuration commands.
        - Support for custom user-defined management commands.
    - JMX-based Management and Monitoring.

GridGain 2.x (and earlier) Release Notes

- Compute Grid
    - In-Memory MapReduce.
    - Pluggable failover
        - Job-stealing failover SPI.
    - Pluggable topology
        - Basic topology SPI.
        - Node-attributes topology SPI.
        - Node-filter topology SPI.
    - Pluggable collision resolution
        - Fifo-queue collision resolution SPI.
        - Priority collision resolution SPI.
        - Job stealing collision resolution SPI.
    - Pluggable job checkpointing
        - JDBC-based checkpoint SPI.
        - Shared file system checkpoint SPI.
        - Amazon S3 checkpoint SPI.
    - Pluggable load balancing
        - Round-robing load balancing SPI.
        - Weighted-random load balancing SPI.
        - Adaptive load balancing SPI.
    - Pluggable event storage
        - In-memory event storage SPI.
    - Pluggable node discovery
        - Multicast-based node discovery SPI.
    - Job Stealing for keeping all nodes equally loaded.
    - Distributed task session.

- AOP-based job execution.
- Support for redundant job mapping.
- Support for partial reduction.
- Zero deployment and Peer Class Loading.
- SPI-based architecture.
- GridGain loaders for automatic blending into any environment (e.g. app servers, web servers, embedded, stand alone, etc…).