



Universal Payments Platform Technology Overview

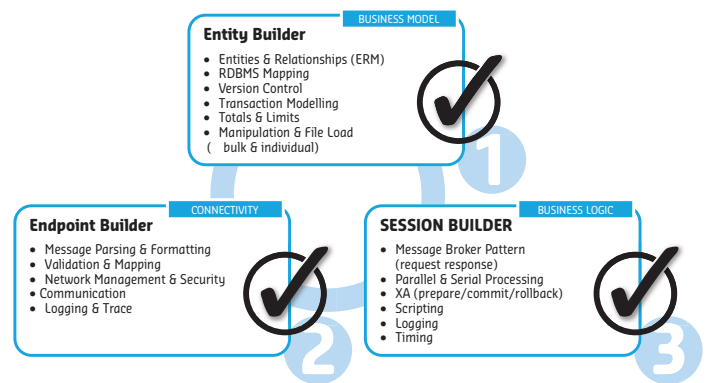
Distra is a global software company that is focused on transforming payments and helping its customers innovate to respond quickly to new payments needs. Through its flagship product, the Distra Universal Payments Platform (UPP), Distra empowers leading banks, payments processors and other companies in the payments chain to quickly address cost, competitive, customer and regulatory drivers by enhancing their existing payments infrastructure. Distra specializes in enabling payments innovation without compromise to performance, security and reliability, with a track record of delivering real-time, 24x7, mission critical solutions in the payments processing space.

These types of mission-critical applications underpin the transaction fabric and are present in most retail banking solutions, card schemes and 3rd party payments processor installations. Distra mainly addresses the tier-1 market involving large processors with substantial processing volumes, and have had solutions deployed in production for more than 5 years without incurring any service level agreement (SLA) penalties.

At the heart of the various pre-built and customised solutions offered by Distra globally sits a piece of core technology that addresses the conflicting requirements of a modern fault tolerant solution. On the one hand the target market demands absolute reliability and a proven track record, whereas on the other hand the world has embraced an agile and highly flexible 'web' like culture that demands responsiveness and adaptability. The Distra UPP marries these two concepts by providing a Java-based service oriented application server, which under the hood integrates strong fault tolerance, real-time performance and efficiency. In other words the Distra UPP looks like a contemporary application server but provides performance and reliability guarantees to the services running on the platform. This piece of the technology is known as MCAS (the Mission Critical Application Server). Some industry experts have characterised MCAS as "providing Tandem like reliability on the Unix server".

The Distra UPP includes a complete GUI based development toolkit as well as a set of pre-built templates that allows a processing solution to be created and configured without any deep technical expertise. The fundamental concepts of a processing solution such as message formats, CODECS, security, communication, database tables as well as the central processing logic is defined using a business oriented terminology instead

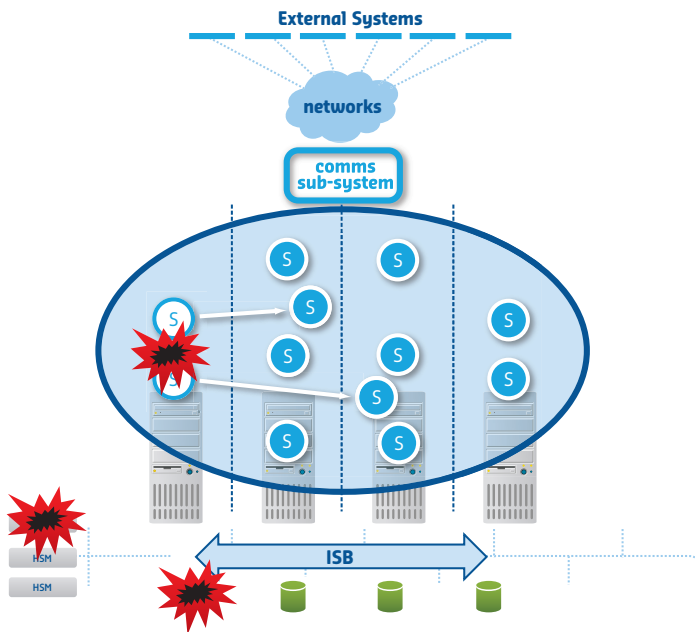
of a technical one. Whereas the Distra UPP has been deployed in traditional EFT scenarios processing core EFTPOS and ATM transactions, it is equally at home in other domains such as real-time ACH and even medical transactions. The Distra UPP



is truly message and logic 'agnostic' in the sense that it has no preconception about the type of transaction or the type of information being processed. The Distra UPP is comfortable supporting the myriad of ISO8583 based protocols with whatever arcane security, communication and CODEC formats it may need, but easily handles contemporary solutions using XML based messaging and WSDL defined interfaces. This approach allows solutions to be designed and implemented in weeks rather than months or years.

The Distra UPP integrates well into existing enterprise architectures given its use of open technology. It provides a WSDL based management interface, will export events and SNMP traps, supports single sign-on and will automatically export any selected database tables in near real-time to a data warehouse, reporting system, portal or other enterprise component.

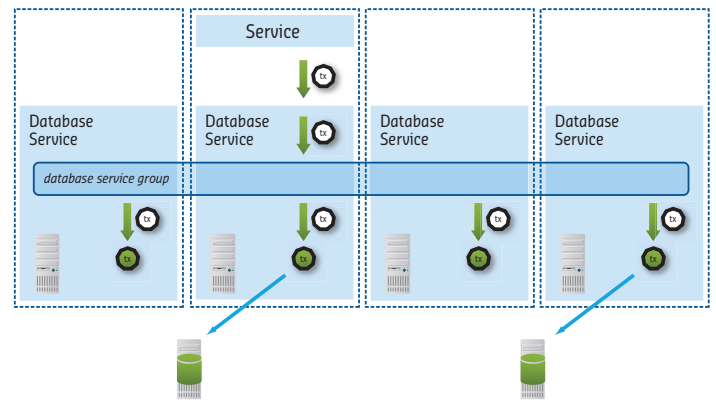
Reliability & Performance



Modern SOA based solutions are often quite elegant and flexible. However, modern real-time, run-forever, no-planned-downtime solutions are quite rare. If these requirements are combined with strict persistence requirements, very high performance, separate geographical sites and maybe a global distribution of processing nodes the list narrows even further. The Distra MCAS platform was designed from the ground up to deliver solutions with exactly these characteristics.

At the core of the technology lies a distributed component model that allows components and services based on these components to run across multiple servers as if they were running on a single server. That is, the application made from these services form a single logical unit even if the services actually are running on different servers. This distributed service model is a key ingredient in the MCAS resilience story. Services are free to move from one server to another in real-time without missing a beat. Relocating services may be performed by an operator to vacate part of the platform for maintenance purposes, or may be triggered automatically if one or more servers fail. This in turn allows the solution to continue processing transactions as long as at least one server remains. It also happens to address the lack of service isolation; one of the main Achilles heels of SOA based architectures. If the Distra UPP detects a problem with a service, it will eliminate the problem by automatically restarting the offending process without interrupting the overall application.

A related feature allows servers to be placed in geographically separate locations. The services can migrate freely between sites allowing processing to continue even if an entire site is down. Transactions are still processed using a single logical application, and information is synchronised and replicated between sites across the WAN.



A less coupled, global solution can be achieved using a federated approach. Since the Distra UPP can be deployed and managed remotely several nodes may be installed and connected via internal endpoints.

Another important MCAS feature is automatic replication of database data. Each site may host one or more databases that act as exact copies of each other. Since all data without exception is written to all databases, the system is highly redundant and can lose one or more databases without interrupting the processing pipeline. Once the failed database(s) come back online, MCAS will copy the missing data to each newly activated database until it is fully up-to-date and redundancy has been restored.

The MCAS philosophy of continuing processing in the face of failure is essential in order to transparently handle outages. MCAS detects abnormal behaviour and automatically excludes the failing part from the pipeline. Obviously the reverse must take place once the part is back and ready for processing (which often presents a bigger challenge). At the heart of the distribution mechanism lays a rigorous consensus mechanism based on virtual synchrony and implemented via a group communication module. This mechanism delivers strong guarantees about the state shared between the services. It also allows database replication to be performed using total order eliminating the needs for exotic transaction constraints such as idempotency.

The rapid increase in processor power over the past decade combined with the efficiency and scalability of Distra's low latency frameworks allows Distra UPP based solutions to process transactions at high rates. The solution is not constrained by having to split the business logic into segregated domains in order to achieve the required performance. Most solutions can run complex entangled logic at rates that satisfy the largest of tier-1 customers.

Software Development

Distra uses an agile software development process, with a high degree of iterative and incremental development. Requirements and solutions evolve between self-organizing, cross-functional teams. The use of continuous integration and a very high degree of automated testing ensures that releases can be cut fairly close to the cutting edge with a high degree of confidence. Since a significant part of the Distra technology is mission critical, Distra uses collaborative development with peer reviews and sign-off for these areas. The PA-DSS certification process requires Distra to track all handling of sensitive data within the code base, which is managed via additional sign-off controls. As an organisation that operates in quite a rigorous and risk-averse market, Distra has managed to strike a good balance between agility and reliability.

The core technology of a Distra solution is naturally based around Java, but Groovy is increasingly being used due to its elegance and power as a scripting language. The key databases are Oracle and DB2, but other databases such as PostgreSQL and HSQLDB are being used.

Development is predominantly done using Eclipse and IntelliJ, but Emacs and vi are also used. Packages such as JUnit, Findbugs and Log4j complement the development tools, as do numerous other Java frameworks.

Build and Integration is performed using JIRA, Bamboo, Maven, Ant and Subversion. JIRA is used in a wider sense for life-cycle and release tracking.

Being a 100% Java solution the Distra UPP can run on virtually any contemporary platform. However, Distra is using and can provide support for Linux Redhat/Suse, AIX and SunOS. Internally both Mac OS X and Linux are used for development and testing.

Please contact one of our offices or visit our website www.distra.com for further information.

Distra Sydney

Level 2 Ballarat House
15 Foster Street
Surry Hills NSW 2010
Australia

P: +61 2 9212 0826

F: +61 2 9280 2075

E: information@distra.com

Distra EMEA

Cardinal Point
Park Road, Rickmansworth
Hertfordshire WD3 1RE
United Kingdom

P: +44 1923 432 605

F: +44 7798 616 932

E: information@distra.com

Distra Singapore

9 Temasek Boulevard
#31-02 Suntec City Tower Two
Singapore 038989
Singapore

P: +65 6559 6253

F: +65 6336 6610

E: information@distra.com

Distra North America

3500 Lenox Road
Suite 710
Atlanta, GA 30326
U.S.A

P: +1 678 250 8418

F: +1 678 264 2250

E: information@distra.com