



AdvOSS Software Quality Attributes

AdvOSS White Paper

For more information Please contact sales@advoss.com

Software Quality Attributes

The difference between an amateur product and a carrier grade product is not much in functionality; it is in Quality. For any serious business to depend on a piece of software to continue to function and evolve as needed, a long list of quality attributes or ‘abilities’ are required. The list seems to be long, but each ability is vital. If you get stuck with something that doesn’t have any **one** of the required abilities, that inability manifests itself in different problematic ways.

To take examples, just imagine working with a product that isn’t scalable? Or isn’t reliable? Or isn’t easily usable? Or isn’t upgradeable? Or isn’t Interoperable with something new that you want to put in? Or isn’t configurable with your network changes? Or isn’t always available? Or isn’t auditable? Or isn’t secure?

It is imperative for operators to look at the products they are purchasing from all the quality attributes that may affect them in future. While most products claim to have similar functionality, it is important for the prudent buyer to ask tough questions about each individual required quality attribute and look under the hood on to the building blocks to see if the product they are buying really has the required quality attributes to deliver on all areas.

Following is a list of commonly expected quality attributes with an introduction on how AdvOSS uses its technology and architecture to achieve them.

1- High Availability

High Availability is the measure of the quality of a software to keep functioning in spite of problems. Since the ‘problems’ can be of many types, different technologies work in tandem to achieve high availability for the overall system. High Availability is achieved through many means.

a- Architectural Means

i. Dispatcher / Server Model

As described in the AdvOSS Technology white paper, all AdvOSS Switching and AAA products work on a Dispatcher / Server model where the Dispatcher distributes work on the Serving nodes. In addition to load balancing, it has high availability benefits as well because when a Serving node goes down, the Dispatcher fails over the load to the remaining Serving nodes without the Clients even knowing the change.

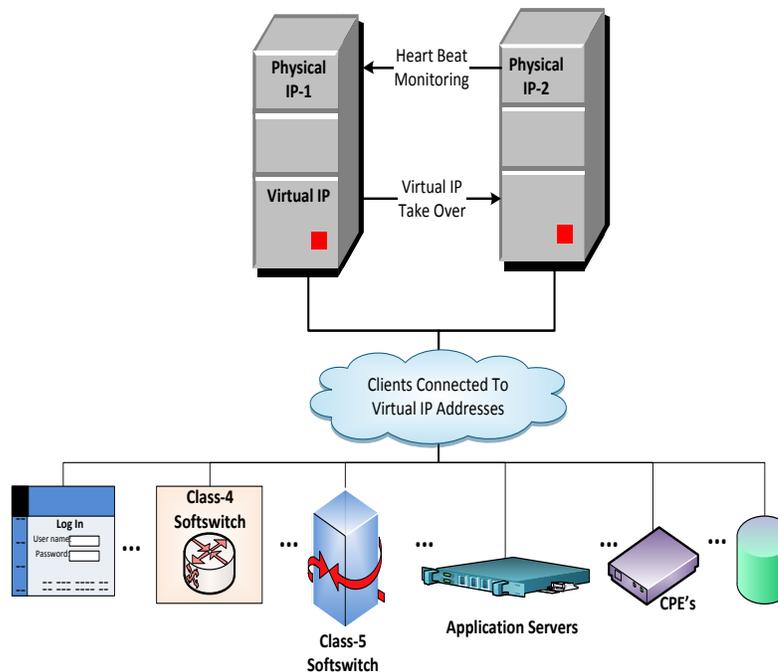
ii. Real-Time Replication

All AdvOSS Databases can be replicated in real-time in multi-master mode. This allows the facilitation in the architecture for any high availability system to use in case the primary or the master database cannot be connected to.

b- Technological Means

i. Heart-Beat Framework

Heart-Beat Framework is an AdvOSS Technology component. It allows one system to become a hot standby for the other system. Working on a combination of Virtual / Physical IPs and a heart-beat monitoring system, Heart-Beat Framework allows the hot standby server to assume the Virtual IP to which all the Clients are connected. This coupled with proprietary state management technology in each product allows the transition to be seamless for the Client who continues to get the Service without interruption. AdvOSS uses its Heart-Beat Framework to provide high availability on the dispatchers.



ii. Primary – Secondary Connections

AdvOSS clients are designed to be able to connect to secondary sources if primary sources are not accessible. This technology along with real-time replication of the data allows a High Availability mechanism whose availability is only dependent on the amount of redundancy in the system

2- Scalability

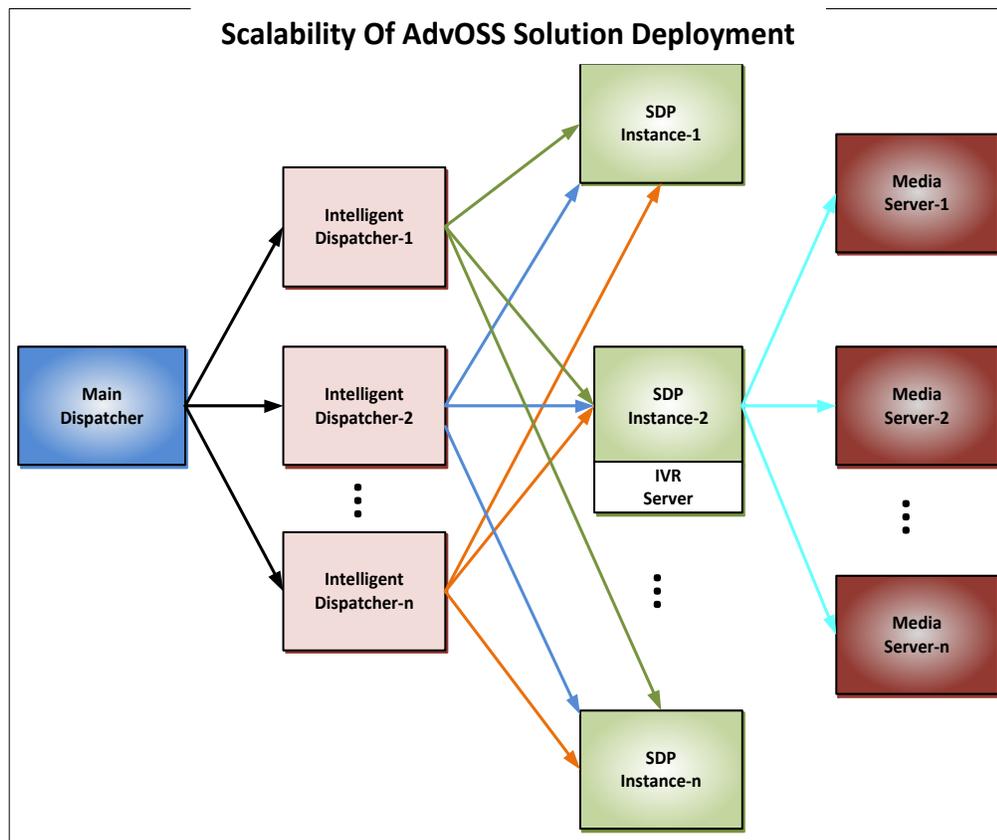
Scalability is the ability of the software to cater for heavier processing loads as the needs arise. In addition to its usual meaning of how heavier loads can be handled, scalability also means 'how small can you start'. Scalable architectures allow both the vendor and the operator to follow a 'Pay as you grow' model. Horizontal Scalability is the ability to be able to add more resources to get higher processing. Vertical Scalability is the ability to be able to add bigger resources to get higher processing.

AdvOSS uses the following technologies and architectures to achieve Scalability

a- Dispatcher / Server Model

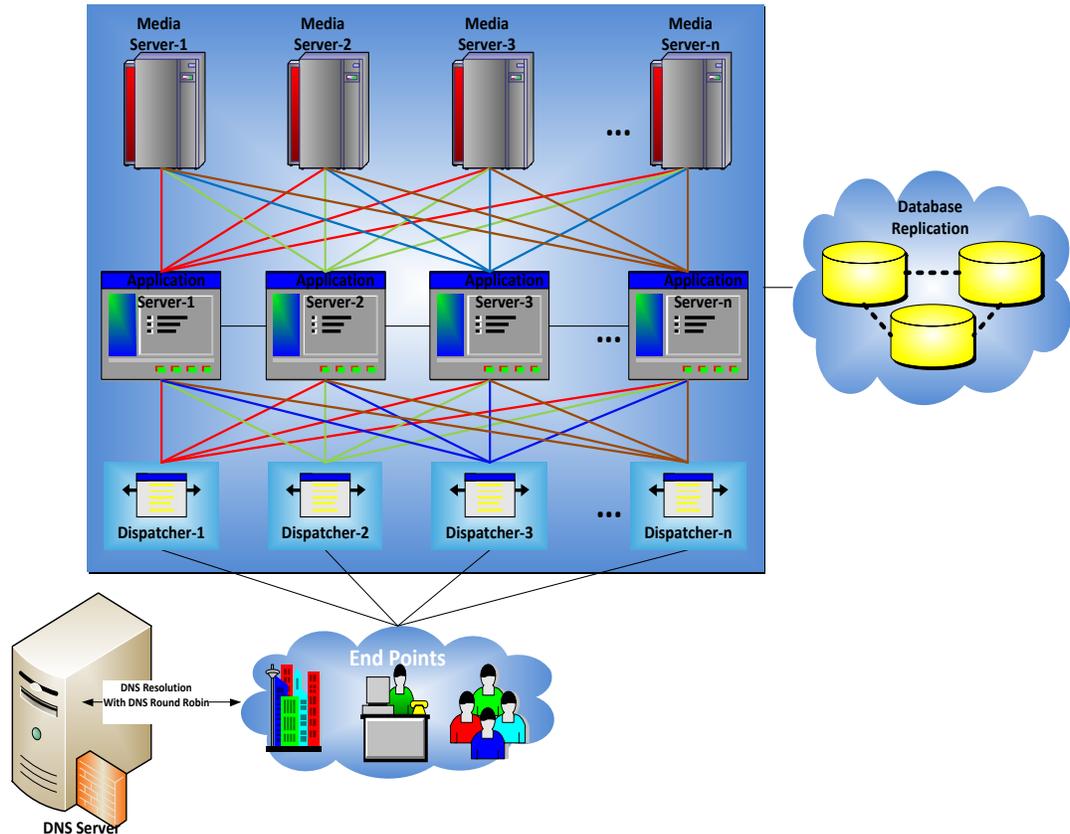
The standard Dispatcher / Server model allows to start very small (usually allowing to start with a single server in 1 + 1 redundancy). As the load increases, processing can be scaled by adding more nodes and shifting the Serving work load onto the Serving nodes. This will allow an Operator to keep on scaling horizontally till the dispatcher reaches its limit of dispatching.

At that time, the Operator can scale vertically to have higher capacity hardware for dispatching with higher RAM and faster CPUs.



b- Layers of Dispatching

The standard Dispatcher / Server model can usually scale enough to support the needs of most Tier-2 Operators. Very large Tier-1 operators may need more scalability. In that case, they need to add an extra layer of Dispatching. Usually it is possible to add Layer 4 Dispatching first which can then serve multiple Layer 7 dispatchers

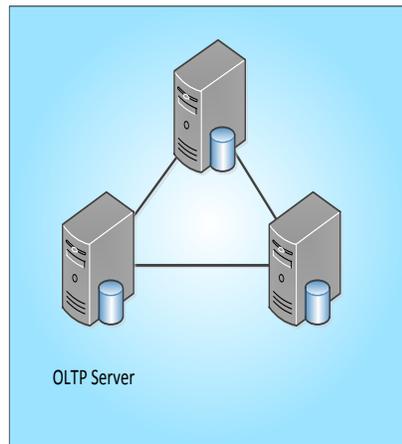


c- Replication

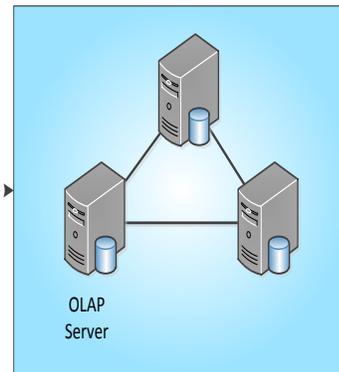
One aspect of Scalability is needed when more and more departments need access to same data and they demand heavier and heavier processing on it. This usually results in a performance hit on the Transaction Processing servers which in extreme cases may become unavailable when some heavy analytical processing is going on.

To cater for this need, all AdvOSS database designs separate the OLTP Servers where all writing of new data is going on from the OLAP Servers where all heavy reads and analytical processing is going on. This provides almost limitless scalability as it is possible to put more and more servers all getting a copy of data in real-time where heavier processing can be done without effecting the master Transaction Processing Servers.

Light weight Servers for Transaction Processing



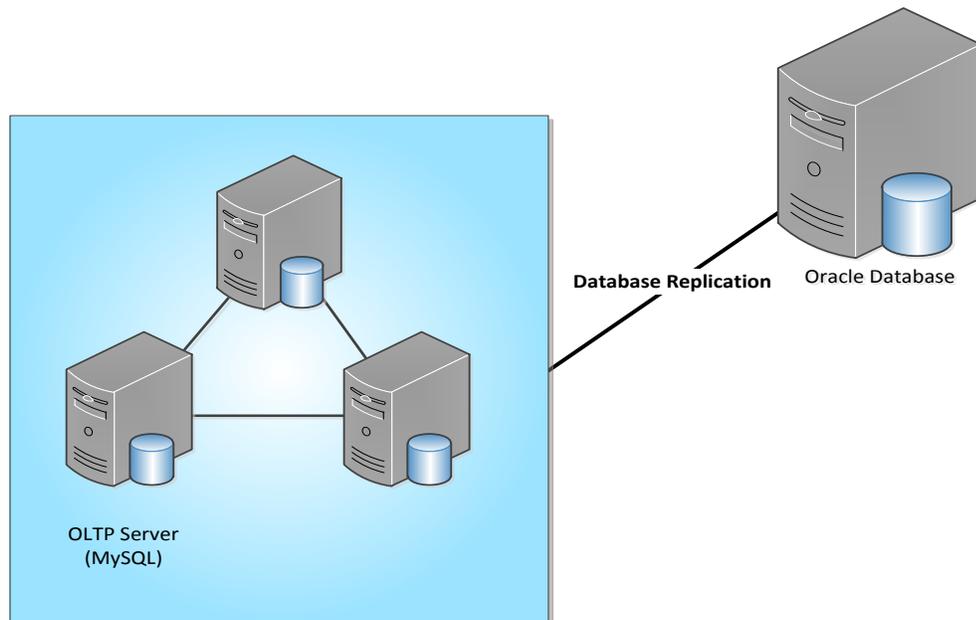
Heavy weight Servers for analytical Processing



Real-Time
Replication

d- Replication between Heterogeneous Databases

When a Department has prebuilt tools for one type of Database, it may insist on getting data in that database and also preferably in real-time. AdvOSS uses advanced database replication middleware to synchronize data between different types of databases e.g. from MySQL to Oracle in real-time.



e- In-Memory Transaction Processing

AdvOSS Database Applications use in-memory transaction processing wherever possible. This greatly increases the capacity of the OLTP servers.

3- Personalizability

Personalizability is the quality of a software product to adapt to the personal needs of the user and accepting the differences in the personality. Personalizability has many dimensions.

a- Personalizability to Organization Structure

The system needs to adapt to your unique organization structure. All AdvOSS Products' user interfaces offer advanced Role Based Security frameworks that allow an Administrator to tailor the rights available to user in fine granularity matching the corporate structure of the organization.

b- Personalizability to Processes

On top of the Application Layer, all AdvOSS Products offer a Work Flow layer which offers 'Process Level' description and control where required. This means that an Operator can always define more complex processes and have attached to the User Interface seamlessly.

c- Localization

All AdvOSS Products offer a GUI where the presentation layer is totally abstracted out from the layers beneath it. Amongst other minor things like allowing a User to change the skin to his needs, it also offers complete Localization for native languages and syntax conventions. A separate Resource File performs translations of master text into what a User sees and allows an Operator to easily translate the master text in his own language.

d- Personalization of the Portals

All AdvOSS Products come with a unified GUI Portal for each 'type of user'. Current portals are:

- Administration Portal
- Customer selfcare portal
- Sales Channel / Franchisee portal
- Vendor Portal

Operator can choose what appears on each portal for all products and for whom.

4- Reliability

Reliability is the measure of how a product behaves in varying circumstances.

a- Heavy Load

Much before the processing capacity is reached at heavier loads; these are the software architectural problems that start manifesting themselves and are a major source of unreliability.

i. Locks:

Most current products run as multi-threaded systems. Many of the reliability problems arise from the dead locks caused by locked shared objects taken in a multi-threaded system. Other reliability problems arise through corruption of shared objects when the right locks are not assumed. AdvOSS has invested heavily into the right middleware called AMPS and builds all of its switching and AAA products over it. AMPS code is written in an Asynchronous way guaranteeing that no locks ever need to be taken. AMPS code has proven to be much more reliable at heavier loads where it keeps behaving in the same way it used to do at lower loads.

ii. Race Conditions:

Race Conditions arise when as a result of multi-threaded execution, a process that started after another one, finishes before it and it is not expected. Such situations give rise to the most subtle bugs in the code which are very difficult to trace and remove. All of the AdvOSS Switching and AAA Products are built over AMPS. AMPS non-blocking execution guarantees that an atomic piece of code is always executed without interruption and this saves many race conditions to appear later.

b- Run-Time Errors

When working against Databases, all AdvOSS Products expose a Transactional API. This API guarantees that all transactions are either completed successfully or rolled back in their entirety when a run-time error occurs and that the Database is always left in a clean state. Even with multiple network elements are involved, the Distributed Transaction Manager in AdvOSS Provisioning Engine guarantees that it is a database that can be 'relied' upon for its integrity of data.

c- Bad Input Data

AdvOSS Products present a Provisioning Layer on top of all Transactions. This layer offers first level of Input sanitation and validates the syntax of all input before it passes it to the Transactional API. The Transactional API checks the semantics and existence of all data and this two layer processing guarantees integrity of data in the database.

5- Extensibility

Extensibility is the ability of the software to be extended beyond the functionality of the original product. AdvOSS uses its technology and architectures in multiple ways to make sure its products remain extensible where required

a- AMPS

All of the Switching and AAA range of products as well as few of the Billing / OSS products are built on top of AMPS. AMPS is undergoing continuous development and this architecture allows all new functionality in AMPS to be available to all the products

b- Work Flow Layers

The Work Flow available in all Switching, AAA and Billing Product ranges allow an Operator to extend the functionality coming out of the box. Simple scripting in sccxml or other supported work flow languages, allow unlimited extensibility

c- Hooks

Deep within the transactions, AdvOSS products provide hooks at all places where new algorithms, rules or work flows may be required to be extended later by the Operator. Any arbitrary extension can be made to the products using these custom programmable hooks.

d- Product Extensibility

Another aspect of extensibility is adding new products. Since all AdvOSS products work against the same information and data model, extending a new product is a seamless experience. When addition of a third party product is needed, AdvOSS Provisioning Engine and its work flow layers provide easy ways to extend.

e- Unified GUI Portals

Each of the GUI portals offered by AdvOSS are modular and allows any product to drop in its menus within the same GUI portal. The GUI is then extended supporting new menu items while still keeping the same security framework in place.

f- Customized Reporting

Its a known that Operators have insatiable appetite for new reports. AdvOSS OLAP Servers allow Operators to extend new reports without effecting the master OLTP Servers. In addition, AdvOSS Support Jasper Reports that gives an intuitive interface to the Operator to write new reports as needed.

6- Portability

Portability is the ability of a product to be ported to other environments mainly other Operating Systems if required. AdvOSS uses the following strategy to keep its products Portable

a- AMPS Based Products

All products developed over AMPS can be ported to any Operating System on which AMPS is ported. AMPS has so far been ported on Linux and Windows. Although our

preferred Operating System for carrier grade installations remains Linux, the possibility of running the products on Windows is open.

b- Java Based GUIs

All GUI Portals are made in the advanced Java framework of Groovy and Grails. The framework can be ported to multiple operating systems.

7- Robustness

Robustness is defined as the ability of a software product to cope with unusual situation. Unusual situations could be many.

a- Attacks

Most common unusual situations are when a system is under a DOS or DDOS attack of some type.

i. DOS Attacks

AMPS provides native admission control and traffic shaping at layer 4. It provides a graceful mechanism to police the traffic and because of its asynchronous nature, can sustain sudden bursts of heavy load as they are processed out at the speed of available processing. This is usually good enough to mitigate many types of DOS Attacks

ii. Distributed DOS Attacks

A Distributed DOS attack is something that is coming from a large number of different hosts and it is not possible to control the attack at mere Layer 4. AdvOSS Products use a Dispatcher at the front end of all products and the Dispatcher is capable of finding out itself if it is under a presumed DDOS attack and exercises 'appropriate layer 7 intelligence' to mitigate the attack. This varies from protocol to protocol but uses the innovative methods to mitigate such attacks in all cases

b- Lost Resources

There are times when a system loses another critical system. Preserving service to customers in such situations may be critical from a business point of view. All AdvOSS Products provide configurable fall back procedures. As an example if the Radius server loses connectivity to the database for an Accounting session, it is programmed to write that activity into a text file that can then be mediated periodically and entered into the database when it becomes available. Even if the Application Server loses connectivity to the Radius Servers it is programmed to write CDRs in text files which can later assure revenue to the Operator.

8- Efficiency

Efficiency is the ability of the software to do the required processing on least amount of hardware.

a- AMPS based design

AMPS provides one of the most efficient execution ways for a piece of code. Written in C and using the most advanced Asynchronous processing techniques, products built over AMPS usually deliver a throughput of 2x to 3x more than competing C based products and 10x to 20x those of Java and other higher level programming languages based products. All of AdvOSS Switching and AAA Range of products enjoy this high level of efficiency as they are all based on AMPS

b- Database Architectures

AdvOSS divides Database processing into separate OLTP and OLAP servers. This allows the OLTP Server to be light weight and run very efficiently. It is typical to be able to do all the transaction processing of up to a million subscribers with a single OLTP Server. OLAP Servers are also divided into two layers. The master servers do all the summary processing and the data along with all the calculated summaries is then replicated to all slave servers making the slave server available for further processing efficiently.

9- Manageability

Manageability is the ability of a software to be managed from different perspectives. Some of them are:

a- Operational Manageability

Operational Manageability is about the ability to manage the product from the perspective of the team running the Operations.

i. Bulk Procedures

In addition to a user friendly GUI, AdvOSS Products work behind a Provisioning Layer. Amongst other things, the Provisioning Layer offer the ability to call any underlying API or Work Flow in Bulk. Ability to run a Bulk script is usually the quickest way to manage an emerging situation.

ii. Roll Back Procedures

Most of the AdvOSS Transactional APIs offer the ability to Rollback the original Transaction. This allows an Operator to simply Rollback if required any damage that was done for whatever reason. The RollBack procedures also allow delayed rollbacks which can be done much later (in fact as late as is possible to rollback). In places where Delayed RollBack procedures are not available for some network elements, AdvOSS Provisioning Engine offers a Transactional Layer that

provides innovative ways for rolling back any work done on the network. All this offers the Operator a simple interface where he is able to RollBack things without even knowing the details of actual Rollback.

b- Monitoring Manageability

Monitoring Manageability is about the ability of the software to be continuously monitored for key aspects in real-time and for the monitor to do critical management over the same Monitoring interface available to him.

AdvOSS products offer built-in SNMP agents that provide the ability to an Operator to not only Monitor their function but also manage things from the same interface when required.

c- Integration Manageability

Integration manageability is the ability to integrate with third party systems. AdvOSS Products expose the same XML based APIs as called from its own GUI to all third party applications making it easy to integrate with any other system. If required AdvOSS Groovy and Grails components can expose the same APIs over JSON or other methods

d- Information Manageability

Information Manageability is about the management of information and data in heterogeneous environments. All AdvOSS products share the same base Information and Data model making addition to new products a breeze and management of information to be done against a single unified model for all products

10- Disaster Recovery

Disaster Recovery is the ability of the software to continue to function when a Disaster occurs. Disasters can be of many types:

a- Natural Disasters

Natural Disasters would constitute fires, earth quakes, tornadoes or other acts of God. Typically the whole data center is expected to be unavailable for the situation to be called a Natural Disaster. For most competing products moving to a Disaster Recovery site is a process that takes hours of down time.

AdvOSS has matured the technology that can replicate data to a geographically remote location in real-time. For AdvOSS, it is just a switching of clients from the core to the DR site and the DR site is ready with latest data up to the last second when disaster occurs. Typical switch over time to a full DR site is of the order of few minutes if not few seconds.

b- Equipment Disasters

There are times, when the hardware and other equipment would fail beyond the amount of redundancy available in the system. AdvOSS products allow periodic backups to be offloaded to a remote site and they can be restored to a point of time just before the time of disaster.

AdvOSS Clients are specially equipped to facilitate recovering from equipment disasters by writing logs into text files which can be mediated and quickly reapplied in case of recovery from an equipment disaster is pushing things to a point back in time. This is specially true for critical revenue sources like CDRs, Voucher Recharges and other financial transactions.

11- Security

Security is the ability of the software to remain protected from unauthorized access. This includes both change access and view access. There are many types of security that need to be present:

a- Intrusion Security

Intrusion Security is about protecting the systems from un-authorized access. AdvOSS uses a multi-dimensional approach to handle Intrusion Security

i. Network Topology

Network Topology goes a long way to protect critical systems from intrusion. All AdvOSS networks are designed on layers of zones and each network element is put in the most protected zone where it can function. The typical zones are:

1- Core Network:

The Core Network consists exclusively of NATed end points not accessible from public Internet. This includes all databases, all Radius Servers, all Application Servers and all Provisioning Servers running in the core.

2- Intranet

Anything that requires access within the organization, can be put on the intranet. This is typical to put the Administration GUI to be available on the Intranet which can only be accessed from within available sub-nets

3- DMZ (Demilitarized Zone)

Only the end points requiring public Internet access are exposed to that sort of access. This will include

For web based access, a Selfcare Portal is specifically designed with minimal code and permissions to be able to call a minimal set of APIs and WorkFlow on the provisioning engine inside the Core

For SIP Traffic, a Session Border Controller is put in the DMZ to exert relevant security controls over all SIP packets entering the network

ii. Web Access Security Framework

Administration Portals are suggested to be on the Intranet, AdvOSS uses Java based Security Framework of ACEGI that provides advanced Authentication and Authorization features.

b- Data Security (Encryption)

The other type of Security is the Data Security.

i. Secured Transport

All AdvOSS Switching and Billing products are built over AMPS. AMPS handles all transport and offers different types of advanced encryption techniques for packets transported over the network. This includes TLS, TTLS and SSL among others.

ii. Encrypted Storage

AdvOSS uses one way encryption for all data that can be stored this way. This would include all passwords, Voucher PINs and other such data. This ensures that even if the security of the data is compromised, this sensitive part of the information is not leaked.

iii. Encryption before transport:

Any data which is highly sensitive, is encrypted well before it is put on the wire. Example is the Voucher PIN Generation Utility that converts all PINs to their MD5 hashes right on the Client machine before it is transmitted to the Server.

All people requiring access to Intranet from public points, need to have VPN connectivity before they are allowed access to Intranet.

c- Forensic Security

Forensic Security is about keep enough logs of all activity to allow to determine compromise or breach later on. AdvOSS ensures that all changes to the data are only done through the Provisioning Engine that keeps detailed logs of IP Addresses, Computer Names, Types of User-Agents and the changes made that can be effectively used in a later forensic analysis

12- Customizability

Customizability is the ability of a software to offer extensions to existing functionality to better suite the needs of a user. Customizability has many dimensions to it:

a- Presentation Customizability

All information that is presented to a user needs to be customized to match its presentation to an internal format.

i. Web Page Customization

AdvOSS uses a skin based presentation layer that allows an Operator to customize the look and feel of all web pages including logos, trade marks and layouts. It uses external style sheets to allow easy changing of fonts, colors and sizes by an Operator

ii. Report View Customization

AdvOSS uses iReport; a tool that allows sophisticated customization of reports including skins, layouts, charts, fonts, images, sub-reports and cross tabs. This desktop based tool allows an Operator to design the look of their reports offline and upload their template files that are then used to generate those reports

iii. IVR Customization

All IVRs played are written in abstracted out SCCXML scripts which can be tweaked by the Operator to customize their flow. New menu items on IVR can be added on the fly. AdvOSS offers a sophisticated Service Creation Framework to allow the Operator create their IVR trees.

13- Flexibility

Flexibility is the ability of a software to adapt when external changes occur. It has many dimensions to it

a- Hardware Flexibility

All AdvOSS Products run on COTS (Common Off the Shelf Servers). This gives an operator the flexibility to put any hardware that it wants as long as it is complying to required specifications

b- Operating System Flexibility

All AdvOSS Products run on a variety of Linux flavors. This gives an operator the flexibility to choose an Operating System for which it may already have in-house expertise.

c- Database Flexibility

AdvOSS products are usually available on at least two common databases; MySQL and Oracle. Operators can choose the database they want and can even choose a mix and match e.g. using MySQL for OLTP servers and Oracle for OLAP servers

d- Process Flexibility

AdvOSS implements all processes on abstracted out work flow layers. Operator has the flexibility to change or tweak those work flows to his needs.

e- Deployment Flexibility

AdvOSS offers multiple deployment scenarios for its products. These deployment scenarios are created to give Operators the flexibility to match the deployment with their rest of the network and system.

14- Resilience

Resilience is the ability of a software to absorb sudden bursts of legitimate loads. AdvOSS uses different techniques of Resilience for different types of loads

a- Network Traffic

AMPS offers an Asynchronous middleware under all of AdvOSS Switching and AAA products. AMPS offers a queued input with configurable Rate Limiting towards the core. This makes sure that if a sudden burst of load comes, it is not absorbed at the edge and is eased off into the core at a rate that the core can take.

b- Web Traffic

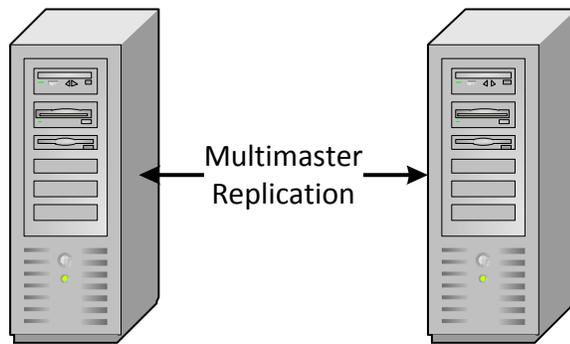
AdvOSS Web portals offer a queueing mechanism to en-queue a sudden burst of requests and to push them into the core on a limited number of configurable connections. This ensures that the core elements (database, Application Servers) are never swamped by requests that they cannot handle.

15- Redundancy

AdvOSS uses different technologies in combination to achieve redundancy in the system and make sure that the redundancy is used towards availability when needed. There are many types of Redundancy needed in the system

a- Storage Redundancy

All databases are redundant and use advanced multi-master replication techniques to keep up to date with each other. This ensures that there is more than one usable copy of all data which is current up to the second. Replication Architectures pose no limit to the amount of Redundancy an Operator may want to have in his system.



b- Computation Redundancy

i. Database

In addition to Storage Redundancy, Multi-Master replication offers computation redundancy as well. Processing can be switched to secondary servers in real-time without loss of continuity.

ii. Switching and AAA

All Switching and AAA architectures work on a dispatcher / Serving node architecture. Dispatcher does real-time load balancing among all Serving Nodes given computational redundancy. If any Serving node goes down, processing carries on through other available nodes.

16- Usability

Usability is the ability of a software to offer its interfaces in a user friendly and elegant way. Usability has many dimensions to it

a- User Friendliness of GUI

AdvOSS uses two types of user interfaces. For all fixed menus, the GUI is hand crafted using advanced web 2.0 techniques making extensive use of Asynchronous calls. For all extended work flows, AdvOSS offers advanced XUI (XML User Interface) based controls to allow Operators to create highly usable GUIs for their own created work flows

b- Response Time of Requests

AdvOSS uses multiple approaches to minimize the latency for all requests

i. Parallel Processing

All AdvOSS work flow engines (xproc, sccxml) are crafted to execute multiple requests in parallel if their nature allows it. This ensures

ii. Read Servers

AdvOSS GUI maintains separate connections to databases for Read and Write operations. For any larger operator, the reads are directed to dedicated read servers instead of write servers and are therefore mostly immune from heavy processing that might be going on in some write servers.

c- Use Case Completion

It is important for the User Interface to provide support for complete use cases of a user. AdvOSS offer a feature rich GUI in the first place. The availability of Work Flow layer on top of the Application layer ensures, that the Operator can add support for use cases not found out of the box.

d- Learnability

Users learn how to use the product over time. But their initial learning curve can be prohibitive and needs extra support. AdvOSS offers content sensitive in-line help on all menu items where the user may find difficulty initially.

17- Accessibility

Accessibility is the ability of a software to be accessible from a multitude of devices and for a number of different types of users. All AdvOSS Products offer multiple interface each of them is configurable by the Operator. A work flow can be attached to each of the methods running behind these interfaces.

- Web based interfaces
- IVR Interfaces
- SMS Interfaces

18- Affordability

Affordability is the ability of a software to keep its total cost within the range of affordability of an Operator. It is mainly about the ability to 'pay as you grow' and It has many dimensions to it

a- Architectural Affordability

This allows an Operator to start with a smaller deployment when its customer base is low. An Architecture that has a higher minimum 'critical mass' to get going, may not be affordable for an Operator starting with a smaller number of customer. AdvOSS has put the pay as you grow model in its core design and the flexibility in its deployment architectures allows the operators to start with the minimum footprint that fulfills their needs and then add more resources as they grow.

b- Licensing Affordability

Licensing affordability is about the licensing model which should not pose any 'minimum' requirements which are higher than the Operator's capacity or immediate needs.

c- Hardware Affordability

AdvOSS uses all COTS (Common Off the Shelf) hardware and avoids using any more expensive proprietary hardware. This allows an Operator to keep its costs in hardware low and also keep its hardware reusable.

d- Network Affordability

Network Affordability is about using more innovative approaches to have the network. AdvOSS built-in support for Cloud and services offered through the Cloud allows an Operator to take maximum benefit of it. The deployment model clearly abstracts out the layers of an Infrastructure provider which can be very affordably arranged in the Cloud

e- Integration Affordability

Integration affordability is about the ability to add more products without extensive integration costs. AdvOSS offers its full range of products pre integrated with each other minimizing or even eliminating all integrations pains and costs associated with adding more products in the line.

f- Total Cost Affordability

Total cost of ownership for a product does not depend only on the hardware and licensing for the system. Major portion of costs is incurred with use of third party software and systems. AdvOSS strives not to require Operators for third party versions and products that they can do without. This is specially true with database licenses where AdvOSS is usually able to start with Standard Editions instead of Enterprise Editions of the same. When and if the Operator grows beyond the capacity of Standard Editions, they can always seamlessly expand to Enterprise editions.

19- Auditability

Auditability is the ability of a software to keep track of what happened, who did it, from where, how and when. AdvOSS offers military grade auditability of its systems. All access to the systems goes through AdvOSS Provisioning Engine if it is modifying any data or through AAA Applications if it is coming from the network. They are both designed to insert a complete audit trail before making any changes. Auditability has many dimensions to it.

a- Who

Each actor to the AdvOSS systems is assigned a unique AdminID and its access passes through Authentication each time they interact with the system. This includes all actors including human beings, schedules jobs, GUI portals, IVR systems, SMSCs, AAA Applications and so on. Since all writes to the database must go through the Transactional APIs, they enforce that auditability information is populated.

b- When

All database activity is logged with timestamps. They are applied by the system automatically and are never taken by the user. Since the detail of each update is also logged, an automatic timestamp is also applied along with storing previous and new values

c- What

The Audit trail stores the details of each Application layer or work flow API that was called. If the request results in the updation of any data, a separate record is always automatically maintained that stores the previous and new values of the record. No record is ever deleted. It is only 'moved' to another deleted records database which is also available online. If the Request results in any changes on the network, they are all logged and attached to the original Request. If the Provisioning Engine is configured this way, it will keep the complete snapshot of the record before making any changes to it.

d- Where

Audit trail not only logs the IDs of authenticated users, it also stores their IP addresses at the time of making the request as well as their user agents and computer names and all other information available in the Request. This usually allows to get more information from Where the request was initiated.

e- How

The Audit trail stores the full Request that came from the Client and the full response to it along with the backend application or work flow API that was called. This is logged at two level. The Request or Work Flow level and at the level of actual network changes.

20- Compatibility

Compatibility is the ability of the software to work with other systems. All AdvOSS products strictly comply to industry standards where possible. All Switching range complies to SIP with all its new standards, all AAA range complies to the Radius and Diameter protocols. All monitoring complies to the SNMP protocol and so on

a- Forward compatibility

AdvOSS products offer hooks for extensions of APIs and Work Flows for extensions of processes. An added advantage of this approach is that the core Application can be upgraded without effecting the extensions made by the Operator. This approach goes a long way in keeping newer versions of AdvOSS products backwards compatible with older versions and makes it easier for Operators to upgrade to newer versions more easily.

21- Traceability

Traceability is the ability of the Software to offer insight into the inner processing when required. A higher level of traceability is required at time of debugging a problem or at times of new interoperability testing. AdvOSS Switching and AAA products offer layers of traceability that can be turned on by a Maintenance engineer. Going from normal error only traces, to warning traces, to activity traces to full verbose tracing, AdvOSS products make it easy for the engineer to be able to see what is going under the hood.

22- Integrity

Integrity is the ability of a software to maintain correctness and harmony among all related pieces of data. AdvOSS delegates the responsibility of integrity to its Transactional API layer which looks at all cases and ensures that it maintains integrity of data by either committing a transaction in full at all places or doing a roll back from all places always leaving the network in a clean state with integrity.

A second integrity verification layer is added through multiple Analytical Processes running in Background. These processes to designed to identify data integrity violations and raise appropriate alarms

23- Modularity

Modularity is the measure of the extent to which software is composed of separate, interchangeable components, each of which accomplishes one function and contains everything necessary to accomplish this. Modularity increases cohesion and reduces coupling and makes it easier to extend the functionality and maintain the code.

a- Functional Modularity

Functional Modularity means how different functions are kept separate from each other in different pieces of code. AMPS goes a long way in keeping functional modularity in AdvOSS Products as it requires all code to be broken into independent non-blocking modules. Each module has absolutely no coupling with any other module. This allows all the AMPS code to be maintained without much concern about coupling with other modules.

Other AdvOSS products and their APIs are also designed to keep them fully cohesive and modular with the API doing everything they need to do within themselves. Wherever possible AdvOSS products use a Message Passing Asynchronous mechanism to communicate between modules in order not to create any coupling between them.

b- Architectural Modularity

Architectural Modularity is about diving the product into separate layers each of them with a separate concern. AdvOSS offers multiple layers completely embracing one concern with them and delegating other concerns to layer above or below them.

The layer starting from the bottom most are:

1. Data Layer
2. Data Modification Language Layer
3. Transactional API layer for Applications
4. Work Flow Layers
5. Provisioning Layer
6. Presentation Layer

24- Maintainability

Maintainability is the ability of a software to adapt to changes, improve over time, correct any bugs and be proactively fixed through preventive maintenance.

AdvOSS facilitates maintainability of products in many ways

a- Monitoring Agents

AdvOSS Products offer monitoring agents within each product that allows any standards compliance manager to monitor them in real-time. They also throw alerts that can be trapped by a manager. Allows the Operator to proactively fix many of the problems.

b- Modular Code

All AdvOSS Code is clearly divided into modules which are cohesive and almost decoupled from the rest of the system. This allows that single module to be upgraded without effecting the rest of the system.

c- GUI Based Configurations

Most of AdvOSS products offer GUI based configuration changes. This allows the Operator to maintain the product easily as network and other systems are undergoing changes.

d- Unified Error Reporting Framework

All AdvOSS Products stick to a unified error reporting framework. Each error is unique across time and space and therefore uniquely identifies the underlying problem. Errors come pre classified to attract the attention of the correct role that deals with such

errors. This unique error handling feature goes a long way in easy maintenance of the product over time.

e- Traces and Logs

AdvOSS Products allow the Maintenance Engineer to turn on different traces and logs at maintenance time that give deeper insight into the code and allows him to fix problems quickly.

f- Redundant Hardware

Redundancy is specially designed to allow the maintenance engineer to isolate a machine from the rest of the system for more detailed debugging. Once done, the machine is put back into the live system.

g- Documentation

AdvOSS offers detailed SoPs, Acceptance Test Documents and short videos to allow the maintenance engineer to follow the given guidelines to achieve his maintenance tasks.

h- Staging Servers

AdvOSS deployment strategies always support a Staging Server, The Staging Server is a live server getting live data from the master servers.

25- Testability

Testability is the ability of a software to be tested thoroughly before putting into production. Although AdvOSS does internal testing before releasing any new versions, they can never be sure to work at an Operator's production systems without testing.

AdvOSS offers a SandBox environment for each of its products. The Debugging Sandbox can be configured against given rules to siphon selected traffic to staging servers for real production simulated testing. This gets very useful in testing the new releases before production.