

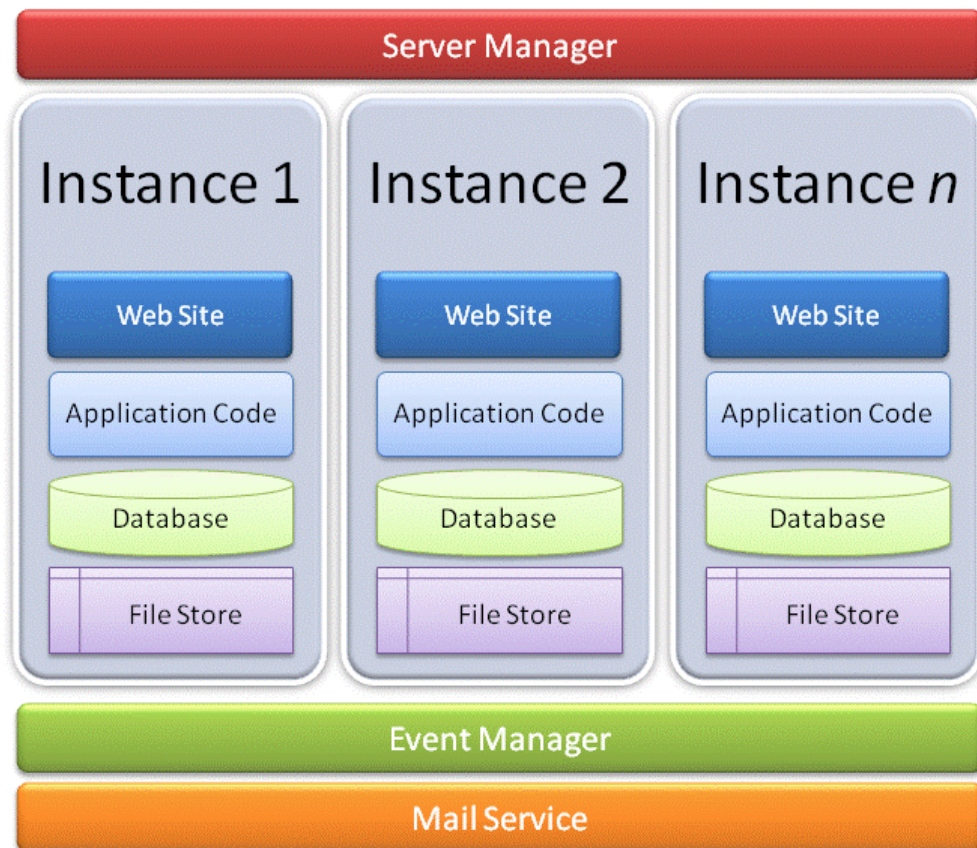


## PPO™ Technical Fact Sheet

This document provides a high-level technical overview of Project Portfolio Office™ (PPO™) including architecture, security, availability, integration and customisation.

### Architecture

PPO has been designed from the ground up to be a highly scalable, web-based, Software-as-a-Service (SaaS) application platform. The diagram below provides a conceptual outline of how a PPO application server is divided into instances with certain shared services that support the instances.



### Software-as-a-Service (SaaS)

A full description of SaaS and its benefits is beyond the scope of this document. In essence however, it means that we can provide the benefits of a fully fledged project portfolio management application to clients of all sizes, at a fraction of the cost of an on-premise solution, without the client having to worry about managing the associated infrastructure. The expertise and economies of scale that we can bring to bear far exceeds that what any single client could achieve with an on-premise installation.



## Instances

Each instance of PPO equates to a client with a unique URL e.g. <http://www.ppolive.com/acme>. An instance will have as many users associated with it as the client requires. Each instance is pretty much self-contained with its own independent file store, database, application code and web site.

Although code is not shared between instances, the code is kept identical across all instances. Going with a multi-instance architecture rather than a true multi-tenant architecture does pose challenges in terms of management but the benefits in terms of security, scalability and availability in our particular circumstances outweighs the reduced costs that could otherwise be achieved. Any or all instances of PPO can easily be moved between alternative servers.

Document storage is an integral part of PPO. Each client's documents are stored in a separate file store for security purposes as well as to facilitate incremental mirroring. Clients are not limited in terms of the amount of space that they may utilise, although there is a general limit of 8MB per document (which may be increased on request). Documents may only be accessed through the PPO application layer.

## Server Manager

In order to ameliorate the higher cost of a multi-instance architecture, a high degree of automation is essential. The server manager provides this automation, allowing for the automated deployment of new instances, rolling out of instance upgrades and monitoring all instances as well as the server itself.

## Technical architecture

Because PPO is a SaaS application platform, the underlying technical details of the implementation is largely irrelevant to clients as it is our responsibility, as the vendor, to take care of this. For the more technically minded, we can however reveal the PPO is based on Microsoft .NET, running on Microsoft 2008 Server R2 and utilising Microsoft SQL Server 2008.

## Security

As with any software application, but even more so in the case of a SaaS application, security considerations play a critical role in the application architecture and the management of the infrastructure.

### SaaS and security

Inevitably, when an organisation's data is kept on servers that are not under its direct control, concerns regarding security and availability of the service come up.

Although this is certainly a valid concern, the bottom line is that in most cases, a service provider such as ourselves, can provide better security than most organisations due to the fact that we are not hampered by having to manage a complex network, with disparate hardware and a multitude of different applications. We only have to protect a single application. Furthermore because of the deep level of expertise surrounding the application and the economies of scale, we can provide a higher level of security than would be available in an on-premise scenario.



### Data center security

Our servers are hosted in high security data centres in South Africa and Ireland, operated by large, reputable hosting providers. More information about the security measures and processes implemented by these providers can be found on their websites:

- South Africa: Hetzner (<http://www.hetzner.co.za>)
- Ireland: Amazon Web Services (<http://aws.amazon.com>)

### Operating system security

The operating system has been configured to provide the smallest possible risk footprint and is regularly checked to ensure that it meets the standards as set by the Microsoft Baseline Security Analyser and other automated scanning tools. This includes ensuring that the latest security patches have been applied, that anti-virus software is up to date and that only the required ports are open. Both physical and local firewalls are in place to ensure that only the appropriate traffic is allowed onto the server.

### Monitoring and active intrusion detection

We have continuous monitoring processes in place to detect potential security threats and to actively lock out IP addresses which exhibit suspicious behaviour (e.g. password guessing, Denial of Service).

### Secure communications

PPO enforces https (SSL) based access across all instances. All https traffic is backed by a 256-bit Thwate digital certificate.

### Application security

The application has been designed from the ground up with security in mind. In addition to logical access control mechanisms which are described in more detail below, specific measures have been incorporated into the application to prevent web based threats such as cross-site scripting, script injection and SQL injection attacks.

A formal security review also forms part of each release to ensure that we have not introduced any features or functionality without considering the security implications.

### Authentication

Authentication of PPO users is done using a standard username and password scheme. PPO provides the ability to automatically e-mail users when they have been added to the system with a system generated password which they will have to change on first login. All passwords are stored in encrypted format on the server. Each instance of PPO can be separately configured to meet the client's specific requirements in terms of password policy, including expiry of passwords, re-use of old passwords, password complexity, and retry counts.

### Authorisation

Authorisation of users is achieved using user groups (which determine what they can do) in combination with data filters (which determine what information they have access to). In addition, custom validation can be implemented to further restrict the ability of users to perform certain actions (see the section on custom validation later in this document).



### **Accountability**

Detailed audit logs are maintained of each users actions to ensure accountability and to provide traceability. These logs are also used by automated monitoring systems to provide information about current activity, usage and to identify anomalous behaviour.

### **Data Privacy**

As per the subscription agreement, all client data is treated as strictly confidential and will never be sold or otherwise wilfully disclosed. All backups are encrypted to protect against accidental or malicious disclosure. The logical separation of instances further mitigates the chance of accidental disclosure.

### **Availability**

An application is only useful to its users if it is available. We therefore have extensive policies, procedures, and automated systems in place to ensure that the PPO application remains available to users.

### **Physical infrastructure**

The first link in the availability chain is to ensure that the physical infrastructure that supports the application remains available. The data centres in which the PPO servers are hosted ensure this availability with the following measures in place:

- Physical security measures
- Resilient and redundant network infrastructure with high-speed connections to the internet
- Climate control
- Uninterruptible Power Supply (UPS) including standby generators
- Fire detection and suppression systems
- 24 hour monitoring and on-site technicians
- Replacement of any faulty hardware within 1 hour
- Hardware based RAID (disk redundancy)

### **Monitoring and response**

The PPO application is continuously monitored from an off-site location using a specialised service provider. If the PPO application does not respond within 3 minutes automatic SMS's and e-mails are sent to multiple support staff who then kick off a response plan based on a set escalation procedure.

### **Client communications**

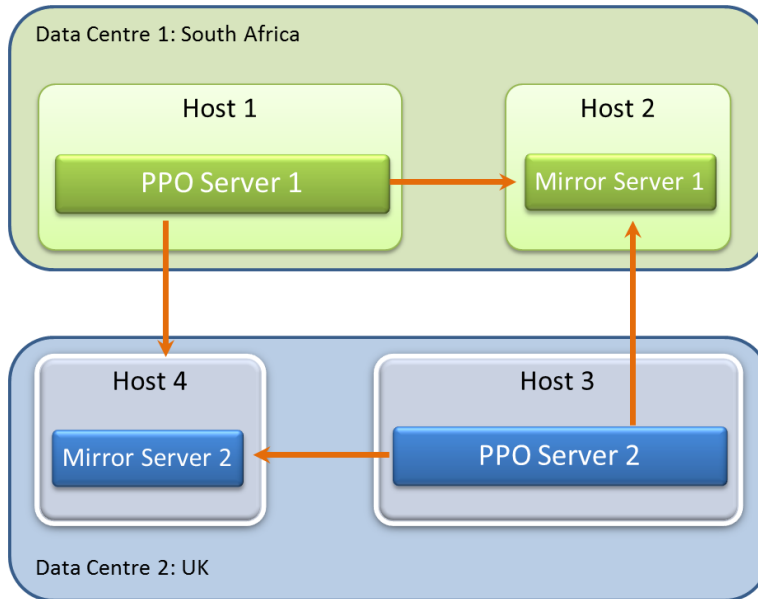
If the PPO application becomes unavailable (whether for scheduled or unscheduled downtime), support staff update our Twitter feed (<http://www.twitter.com/ppo>) to ensure that users are aware of the outage and are kept abreast of the latest developments. If the outage lasts for longer than 30 minutes (during business hours), additional e-mail based communications will be sent out depending on the nature of the incident.

### **Mirroring**

Mirror copies of all client data are maintained on specialised mirror servers which are updated every hour. Each application server has both an on-site mirror (same data centre, different server) as well as an off-site mirror (different data centre, different server). All client data is encrypted while in transit.



The diagram below shows the mirroring configuration for two PPO application servers:



**Backups**

In addition to the mirroring process described above, historical, snapshots are also maintained for additional redundancy.

**Fail-over and disaster recovery**

In the event of a loss of a server or data centre, whether as a result of hardware failure, power failure or communication failure, we have a comprehensive fail-over process in place.

We make extensive use of pre-configured virtual servers, which allows us to provision a new PPO application server within minutes in any one of our data centres. In conjunction with the mirror process previously described, this allows us to move any or all clients to an alternate server, data centre or hosting country within a very short period of time.

This process is continuously tested as part of disaster recovery preparedness but is also used routinely when upgrading our hardware or doing load distribution.

**Customisation**

In most cases, the standard PPO functionality is sufficient to meet clients’ requirements. PPO can be extensively configured through the front-end to fit a specific client’s business and project management methodology.

In some case however there is a requirement for customisation that cannot be achieved with the standard functionality. The following customisation can be done for clients with and installed user base of more than 50 users:



### Custom entities

In addition to the standard entities (such as issues, tasks, etc) that are available in PPO, we also allow for up to 3 custom entities to be configured which work in exactly the same way. It is up to the client to decide what they want to use these entities for and how they want to configure them. Some examples of custom entities include Lessons Learnt, Benefits and Balance Scorecards.

### Custom reports

PPO has a wide range of standard reports which meet most client requirements. However in some cases a client may require a slightly modified report or requires a totally new report. In this case, we can change the configuration of an existing report or develop a totally new report to meet the client's requirement. All custom report development is charged for on a time and materials basis.

### Custom validation

PPO performs limited validation when adding or updating items such as projects, issues, employees, tasks, etc. through the web front-end. This validation is however generally limited to ensuring that the field values submitted conform to the field type for each value (e.g. ensuring that only numeric values are added in a numeric field or that the length of a text field does not exceed its maximum length).

In some instances however, more sophisticated, rule based validation may be required. In addition to the standard validation rules that can be configured by end-users through the PPO web front-end, custom validation rules can also be set up for clients on request.

Examples:

The implementation of custom validation has been done in a way that allows for a lot of flexibility. The following are some examples of what can be accomplished with custom validation, broken up into natural categories:

#### Range checking

- The 'Budget Amount' must be greater than 0 but less than 5 000 000.
- The 'Actual Start Date' for a task cannot be in the future.

#### Field relationship checking

- The 'Planned Task End Date' must be on or after the 'Planned Task Start Date'.
- The 'Status' of a task, cannot be set to 'Complete' unless the 'Actual % Complete' is also set to 100%.
- If a scope change is marked as approved, the 'Approval By' field must be provided.

#### Security related restrictions

- The 'Project Status' can only be set to 'Complete' by a member of the 'Project Office' user group.

#### Complex business rules

- Only one cost item can be added for a particular 'Financial Year' and 'Cost Category'.
- There may only be one task of type 'Project Task' per project.



### Complex calculated fields

One of the standard field types available in PPO is a calculated field and in most cases this can be configured by the system administrator or implementation staff. In some cases however the calculation is so complex that it cannot easily be implemented using the standard functionality. In these cases, we can implement a custom database function, which is referenced through the standard calculated field.

Examples:

- Calculating a project health RAG based on complex business rules and information found elsewhere in PPO.
- Calculating the age of an item (e.g. an issue) with reference to the date and time on which it was originally logged.
- Calculating Balanced Scorecard values using a custom entity and a complex set of business rules.

### Custom events

When event queuing is enabled for a PPO instance, every time that an entity (e.g. a project or and issue) is updated, an event is triggered and placed on the event queue. These events can then be used to trigger certain actions by implementing appropriate custom event handlers. Refer to the Integration and Interoperability section later for more information about the type of event handlers that can be implemented.

This functionality can be used to implement various kinds of notifications, workflows as well as for integration purposes.

Examples:

- When a project is added, automatically add a set of health indicators to the project based on the type of project (e.g. Large Projects vs Small Projects).
- When the "Submit for Approval" checkbox is checked on a project, automatically create an approval and lock down the project against future editing until the project has been approved.
- When the status of a task is set to 'Complete', send an e-mail to the project manager (rule base e-mail based notifications).
- When project information is updated on PPO, then update certain information on a third party application by calling a web service.

### Custom web services

Custom web services can be implemented to achieve complex workflow scenarios or for integration purposes. These web services are similar to the standard web services offered by PPO (refer to the section on integration and interoperability). They are usually used in conjunction with custom events to achieve the business objective.

### Report mailing

Any report within PPO can be mailed to a pre-set group of recipients. PPO allows reports to be mailed to either the members of a user group or a list of employees (defined using an employee filter). In addition, the PPO report mailer allows the content of the report to be varied for each recipient (e.g. on a weekly basis, each user should receive a report of the tasks that have been assigned to them).



## Integration and Interoperability

In order for PPO to integrate with other client applications, it provides various mechanisms for retrieving and updating information programmatically. The three main mechanisms for integrating with PPO are web-services, e-mail based integration and the asynchronous event handler.

### Web services

PPO has a collection of web services that come standard as part of any PPO instance. These web services can be called to retrieve information for use in a third party application or it can be used to update information in PPO based on information from a third party application.

These web services are available at no cost to all clients, subject to reasonable use restrictions. All web services are secure and require the caller to first authenticate itself before being able to perform any retrieval or update of information.

For a full list of the available web services, you may browse to the following address: <http://www.ppolive.com/demo/webservices/integration.aspx>. Web service based integration is best suited to synchronous (i.e. real time) integration where the client has sufficient skills to implement the business logic around the integration.

We are also able to implement custom web services to meet more complex integration scenarios. These custom web services can be developed for clients on a time and materials basis.

### Event Management System

Whenever an event occurs within PPO, such as when an item is added, updated or deleted, PPO places a message on queue. The Event Management System monitors this queue and determines whether an event handler has been set up to deal with the particular type of event. If an appropriate handler is found, it calls it and passes it the information pertaining to that event. Currently, there are four different types of event handlers that can be set up, namely:

*Application event handler:* This type of handler calls a command line application and passes it the information pertaining to the event as an argument in XML format. The command line application is usually a custom developed application that does the actual work required for the particular integration scenario.

*Mail event handler:* This type of handler sends an e-mail based on the event. The information in the e-mail as well as the intended recipients is determined by the business rules implemented in the event handler.

*Web service handler:* This type of handler calls a PPO web service, which can either be a standard or a custom web service. This type of handler is useful for implementing complex workflows or where complex business logic is required in order to update information on PPO.



### **Inbound E-mail based integration**

For batch or near real-time, inbound (i.e. something will be updated on PPO) integration, e-mail based integration is usually the simplest to implement. All that needs to be implemented on the client side is to send an e-mail with an XML attachment containing the relevant information to a designated e-mail address.

The mail management system within PPO contains a feature by which it can be instructed to monitor a specific mailbox account. When mail gets delivered to that mailbox, it places the message on a queue, from where it will be handled by The Event Management System as described above.

### **Example 1: Integration with HEAT (a call logging system)**

The client uses HEAT as a call logging system. Whenever a call is created on HEAT, a corresponding issue must be created on PPO and assigned to a 3<sup>rd</sup> party contractor to address. If the call is updated on HEAT the issue on PPO should also be updated and vice versa.

Since both HEAT and PPO support e-mail based notifications and updates, the integration was accomplished by HEAT sending an e-mail to a PPO mail box as soon as a call is logged or modified. An event handler was then set-up on PPO which creates or updates the issue on PPO and assigns it to the relevant person. Similarly, if the issue is updated, an event handler sends an e-mail to the HEAT system in the prescribed format, resulting in it being updated on HEAT.

This type of integration can be used in a wide variety of scenarios, including those where the systems to be integrated are on disparate networks, or not always available on-line.

### **Example 2: Integration with SAP**

The client uses SAP PS and wants to update project cash-flow forecast in SAP PS from PPO.

A web service was developed in SAP XI, which is called by an event handler that was set up on PPO. Whenever project costs are updated, the event handler calls the web service, which in turn updates SAP PS.

### **Example 3: Integration with a custom web-application**

The client has a simple, custom developed web application which shows key milestones per project and allows users to update the % complete of the milestones. The requirement was for the web application to show the projects and milestones from PPO and to update the relevant tasks on PPO in real-time.

The web application simply calls a web-service synchronously on PPO to get the list of projects and milestones and displays it to the user. If the user updates the % complete, another web service on PPO is called, which updates the task with the % complete.

### **Example 4: Providing information to a data warehouse**

The client wanted project risk information to be fed to its data warehouse on a daily basis.



A scheduled task was created, which calls a script developed in C# that pulls the required risk information from PPO via the API. It then builds a text file from this and FTP's the file to the warehouse staging area.

## Client requirements

### Browser requirements

In order to access PPO, any standard web browser can be used, including Internet Explorer, Google Chrome, Mozilla Firefox and Apple Safari. It is recommended that the latest or at least a fairly recent version of the browser be used for the best experience. It is further recommended that the client computer should have a screen resolution of at least 1024 x 768.

### Network requirements

PPO's bandwidth requirement is fairly modest and is comparable to normal internet browsing. However due to the fact that there are very few images in PPO as well as the fact that we use compression, the bandwidth is typically even lower than normal web browsing.

For the purposes of network administrators who need to assess the impact of PPO usage on their internal bandwidth, we typically use a figure of 0.6MB per licensed user per day for inbound traffic (server to client). Outbound traffic is typically about 15% of that or 0.09MB per day. For a detailed FAQ about PPO's bandwidth requirement, please contact the PPO support desk.

**For assistance and more information relating to this factsheet please contact us at [support@ppo.co.za](mailto:support@ppo.co.za).**

### Post Vision Technology (Pty) Ltd

**Tel:** +27 (12) 348 2366  
**Fax:** +27 (12) 348 8382  
**Email:** [support@ppo.co.za](mailto:support@ppo.co.za)  
**Web:** [www.ppo.co.za](http://www.ppo.co.za)  
**Physical Address:** Ground Floor, Building 3, Ashlea Gardens Office Park  
Cnr. Garsfontein & Matroosberg Roads, Ashlea Gardens, Pretoria  
**Postal Address:** PO Box 75760, Lynnwood Ridge, 0040, Pretoria, South Africa