

Project HealthDesign Common Platform FAQ

Basics	2
What is the Project HealthDesign Common Platform?	2
What services are provided by the Common Platform?	2
Why did Project HealthDesign develop the Common Platform?	3
How do personal health applications use the Common Platform?	3
What programming languages are supported by the Common Platform?	3
Can I use the Common Platform now?	3
Is there a hosted demo version of the Common Platform server available?	4
Are the Common Platform server source code or binary files available?	4
Architecture	4
What programming language is the Common Platform written in?	4
What server components are required to run the Common Platform?	4
Authentication Service	4
What functionality does the Authentication Service provide?	4
How do I generate an Authentication Token?	4
What is a SecurityTokenUniqueID?	4
What is a Nonce value?	5
How do I generate a ProofToken?	5
How long is a SecurityContextToken valid?	5
Registry Service	5
What functionality does the Registry Service provide?	5
What is the difference between a User account and a Patient Account?	5
Can I log in with a patient account?	5
How do I retrieve a list of patients?	6
Access Control Service	6
What functionality does the Access Control Service Provide?	6
What are User Roles?	6
What are Access Control Rules?	6
Are there any default Access Control rules?	7
Observations Service	7
What functionality does the Observation service provide?	7
What types of observations are supported by the Observation Service?	7

What is the difference between loosely-typed observations and strictly-typed observations? _____	7
Medications Service _____	8
What functionality does the Medication Service provide? _____	8
What kinds of medication information are supported by the Observation service? ____	8
General Information _____	8
Are multimedia attachments supported by the Common Platform? _____	8
Can Common Platform records be annotated? _____	8

Basics

Q: What is the Project HealthDesign Common Platform?

A: The Common Platform is a set of software components that provide common, shared functions to a variety of personal health applications (PHAs). The functions include data storage for medications and patient observations, as well as mechanisms for authenticating users and securely sharing patient data. The common platform is currently implemented as a set of web services that are accessible over the internet via standard interfaces (WSDL). PHAs may use the defined interfaces to receive services from one or more of the common platform components, as needed. The goal of the common platform components is to reduce implementation time and increase interoperability for PHAs..

Q: What services are provided by the Common Platform?

A: The following five services are implemented and supported in the common platform:

Component	Description
1. Registry Service	Stores demographic and password information for the users and applications that may access the common platform, as well as for any patients whose data are managed by the platform.
2. Authentication Service	Authenticates the identities of users and applications that wish to access the common platform, and provides single sign-on across all of the platform components.
3. Access Control Service	Allows patients to control the sharing of their health information very selectively. Stores the rules that specify which resources and operations PHAs have access to. Enforces these rules when users request read and/or write access to specific patient data.
4. Medications Service	Stores and makes available the list of medications that patients take regularly. Relies on the Authentication and Access Control services to control access.
5. Observations Service	Stores and makes available health-related observations that are captured

	by or on behalf of patients outside of health care encounters. Relies on the Authentication and Access Control services to control access.
--	--

Q: Why did Project HealthDesign develop the Common Platform?

A: The nine grantees of Project HealthDesign expressed a very specific set of functional requirements for the storage and sharing of personal health data (see [Functional Requirements](#)). A rigorous analysis of available health data standards and PHR platforms (including Google Health and Microsoft Healthvault) revealed that no existing resources fully meet those requirements (see [Comparative Analysis](#)). Project HealthDesign developed the common platform both to provide its grantees needed authentication, data-storage, and data-sharing services and to explore the optimal design and implementation of a common platform for PHRs. The learnings from this process can inform the development of PHR architectures, particularly those that distinguish between personal health applications and shared platform services. Additionally, the common platform software itself may be made available in the future to other projects that need similar resources.

Q: How do personal health applications use the Common Platform?

A: PHAs access the common platform via web-services requests made over the internet. The Common Platform has an interface specification written in the standard web services definition language (WSDL), which defines all of the interfaces and methods supported by the common platform services. This WSDL file can be used to automatically generate client “stub” code, which applications call to connect to and interact with the Common Platform. Software to generate client code from a WSDL file is available for most major programming languages.

Q: What programming languages are supported by the Common Platform?

Because the Common Platform is a SOAP-based web service, defined by a standard WSDL specification, any language that supports the SOAP messaging framework may be used to communicate with the services. Existing Common Platform client PHAs have been written in Java, C#, and PHP.

Q: Can I use the Common Platform now?

A: Currently, access to the Common Platform web services is limited to the Project HealthDesign grantees. However, Project HealthDesign is discussing whether and how to make the services and/or the underlying software code available to others. In the meantime, interested parties are encouraged to review the functionality of the common platform to see whether some or all of the platform components may be useful to them.

Q: Is there a hosted demo version of the Common Platform server available?

A: A hosted instance of the Common Platform has been made available to the grantees of Project HealthDesign. Access to the hosted service is not yet available to those outside of Project HealthDesign.

Q: Are the Common Platform server source code or binary files available?

A: Not at this time. Currently, Project HealthDesign is evaluating what would be required to release the Common Platform as an open-source solution. Further information will be forthcoming.

Architecture

Q: What programming language is the Common Platform written in?

A: The Common Platform is written in Java EE 5.

Q: What server components are required to run the Common Platform?

A: The Common Platform must be run within a Java Application Server, such as Glassfish. The server uses a MySQL database for data persistence.

Authentication Service

Q: What functionality does the Authentication Service provide?

A: The Authentication Service provides “single-sign-on” functionality for all of the Common Platform services. After providing valid login credentials, the Authentication service produces a temporary unique token that is shared with the PHA and made available to all other connected services. The client PHA then submits this temporary token, in addition other authentication information, when making subsequent requests to the Common Platform Services.

Q: How do I generate an Authentication Token?

A: Once the Authentication Service has validated a user’s credentials, the service returns a SecureContextToken element that consists of a SecurityTokenUniqueID and a ProofTokenEcrptionKey, in addition to other identifiers. When making subsequent requests to a Common Platform, the client PHA must provide an Authentication token. The Authentication element consist of the SecurityTokenUniqueID, a Nonce value, and a ProofToken

Q: What is a SecurityTokenUniqueID?

A: The SecurityTokenUniqueID is the value that was returned by the Authentication Service after successful authentication. This value is the unique identifier used by the Common Platform services to look up the SecureContextToken stored by the Authentication Service.

Q: What is a Nonce value?

A: The nonce value is a client-generated string. Any value will suffice at the present time but a unique time-specific value is preferable (e.g., the current date + some identifier).

Q: How do I generate a ProofToken?

A: To generate a ProofToken, a client PHA must concatenate the Nonce value that was generated by the client with the ProofTokenEncryptionKey returned by the Authentication service. The resulting value must then be encrypted using the SHA-1 encryption algorithm and converted to a hexadecimal value. Most programming languages have utilities for SHA-1 encryption and hexadecimal encoding.

Q: How long is a SecurityContextToken valid?

A: A SecurityContextToken is maintained by the Authentication service for 24 hours from when it is first created. After the SecurityContextToken expires, the client PHA must resubmit the user credentials and obtain a new SecureContextToken.

Registry Service

Q: What functionality does the Registry Service provide?

A: The Common Platform Registry Service provides the ability to store, retrieve, update and delete Patient, User, and Application information.

Q: What is the difference between a User account and a Patient Account?

A: User accounts are used to log into the common platform. Each user account is associated with a username and password. Only a single person should be associated with a single user account. No personal health information is directly associated with a user account.

A patient account, on the other hand, is tied directly to the personal health information stored by the Common Platform components. . However, a user account may be linked to a patient account via the Access Control Service, to denote that a patient account stores the health information for a specific user. Additionally, a patient account may be associated with multiple other user accounts to allow patients to share their data with family members, physicians, etc.)

There is a many-to-many relationship between user accounts and patient records.

Q: Can I log in with a patient account?

A: No, only user accounts may be used to authenticate and log-in to the Common Platform. In fact, patient accounts have no username or password elements associated with them. User accounts may be associated with a patient account via the Access Control service. The user account can then be used to

log in and update personal health information related to the patient record, as long as the access control rules permit.

Q: How do I retrieve a list of patients?

A: To retrieve a list of patients, you make a call the `getPatientRecordsSimple` method of the Registry service. The client PHA must provide the last name of the patient when the request is made. The Registry service will respond with a list of all the patient records that the authenticated user has read access to. At the present, there is no method available to request a specific patient by the unique patient ID.

Access Control Service

Q: What functionality does the Access Control Service Provide?

A: The Access Control Service allows patients to control the sharing of their health information very selectively.. The Access Control system stores User Roles and Access Rules. The combination of these two concepts determines what kind of access a user may have to various types of data in a patient record.

Q: What are User Roles?

A: User Roles establish a relationship between a user record and a patient record. Roles are pre-defined within the Access Control service and include Record Custodian, Healthcare Provider, Physician, Parent, etc. Roles are hierarchical in nature such that a higher level role (e.g., Healthcare Provider) subsumes lower level roles (e.g., Physician, Physical Therapist, etc.). Operationally, this means that a user with a higher-level role has all access privileges assigned to any subsumed role. Roles are used when constructing Access Control rules. It is crucial to note that a user role is only relative to a given patient record. The concept of a system-wide role (e.g., a physician role relative to all patients) is not supported by the access control system.

Q: What are Access Control Rules?

A: Access Control Rules determine the kind of access to a patient's personal health information a user account may or may not have. Access Control Rules are made up of the following elements.

- Patient ID – The patient that the rule pertains to.
- Role – The type of user role (specific to the patient) that the rule pertains to.
- Resource – The kind of patient information (e.g., medications, observations) the rule pertains to.
- Operation – The action that the rule pertains to (e.g., read, update, create, delete, etc.)
- Context – The Application that the rule pertains to (data in the common platform may only be accessed via applications).
- Access – Indicates if the rule grants access or denies access.

The Role, Resource, Operation, and Context fields are hierarchical in nature. For example, with the Resource Field, access may be granted at a general level (e.g., granting access to all Observation Records) or at a fine-grained level (e.g., granting access only to Physical Activity observations, a type of Observation record).

Q: Are there any default Access Control rules?

A: Yes. When a user first creates a Patient record, that user is assigned a role of “RecordCustodian” with respect to the new patient record. An access control rule is created for the patient record granting all access to all resources for the Record Custodian. With these defaults, the user record may now add other user roles and access control rules and roles for the patient record, as needed.

Observations Service

Q: What functionality does the Observation service provide?

A: The Observation Service provides methods to store, retrieve, update and delete observation data captured in the course of daily living. Observations can be as simple and atomic, such as a text comment or journal posting, or complex and highly structured, such as a blood glucose reading.

Q: What types of observations are supported by the Observation Service?

A: The following is a list of the Observation Types supported by the Observation Service:

- GeneralObservation
- HealthcareEncounter
- JournalEntry
- MealOrSnack
- MedicationAdministration
- Pain
- PhysicalActivity
- SignOrSymptom
- ObservableParameter

Q: What is the difference between loosely-typed observations and strictly-typed observations?

A: The observation supports many kinds of observation types. Some are strictly typed and some are loosely typed. Strictly typed observations are those, such as “PhysicalActivity” or “MealOrSnack”, that are intended to store a well defined set of parameters applicable to the type of observation being recorded. Loosely-typed observations, such as “GeneralObservations” and “ObservableParameters”, are ones that are more flexible and can be used to store a wide variety of information. With loosely-typed observations, the type of observation is defined by the data stored in the record. For example, an Observable Parameter may be used to store blood glucose data and heart rate data.

Medications Service

Q: What functionality does the Medication Service provide?

A: The Medication Service provides methods to store, retrieve, update, and delete medication list information.

Q: What kinds of medication information are supported by the Observation service?

A: The following kinds of Medication records are supported by the Medication Service:

- Prescription
- DispenseRecord (for a drug that was dispensed by a pharmacy)
- AdHoc (e.g., over the counter medications, vitamins, etc.)

General Information

Q: Are multimedia attachments supported by the Common Platform?

A: Yes. The Medication and Observation services both allow encoded binary data to be attached to stored records. Attachments may be video photos, audio, text, etc. (all MIME types are supported). When a record that includes attachments is retrieved, the record includes meta-data about each attachment (the attachment size and a file descriptor). A subsequent call may be made to retrieve the attachment data itself (which may be quite large).

Q: Can Common Platform records be annotated?

A: Yes. The Medication and Observation services both allow text annotations to be associated with a record. These annotations can be used to comment on a result without changing the actual value of the result. Special access control rules can be created to allow users to annotate a record but not otherwise edit the record. Annotations are text-only.