



The platform with more opportunities
Invitation to Adventure

bada Overview



What is bada?

bada, (bada, the Korean word for “ocean”,) is a new smartphone platform that allows developers to create feature-rich applications that elevate the user experience in mobile spaces.

User-interactive

Samsung bada includes a new UI framework that supports the next generation Samsung touch UI. The main UI delivers simplicity and ease, without decreasing usage efficiency. The UI framework introduces an open-ended evolutionary innovation from the current touch UI to leverage better user experiences.

Some additional UI tools include the ability to embed the Adobe® Flash® Player and the WebKit Internet browser directly into native bada applications, allowing seamless integration of premiere UI technologies. Also, the bada map control is easily embedded in your applications to deliver interactive mapping and routing with POI features for your users.

The bada platform provides user interaction modalities including motion sensing, fine-tuned vibration control, and face detection. These interfaces open up new possibilities for greater creativity and user interactivity in developing applications. Samsung bada also provides a mechanism to develop sensor-based, context-aware applications. With sensors such as acceleration, magnetic, tilt, weather, proximity, and activity sensors, application developers can easily implement context-aware, interactive applications.

Service-oriented

Samsung bada introduces several new service-centric capabilities that differentiate it from conventional mobile operating systems. These new features include social networking, device synchronization, content management, location-based services, and commerce services, all supported by back-end bada servers. Developers can now create service-centric applications without additional effort compared to developing mobile standalone applications.

Social networking applications can manage user profiles, including buddy information for application users, and can integrate users on most popular social networking services. Device synchronization applications can take advantage of PIM and contacts synchronization between bada devices and servers. Content management applications can store and search contents on devices and servers. Location applications can use landmark stores, maps and geo-coding services. Commerce applications can sell products, such as game items in applications.

What is bada?

Open and Configurable

Samsung bada enables developers to take full advantage of mobile device capabilities to easily create compelling applications. Applications can use device functions to make phone calls, send messages, or access the contact list, as various application controls allow developers to create richer, more consistent applications.

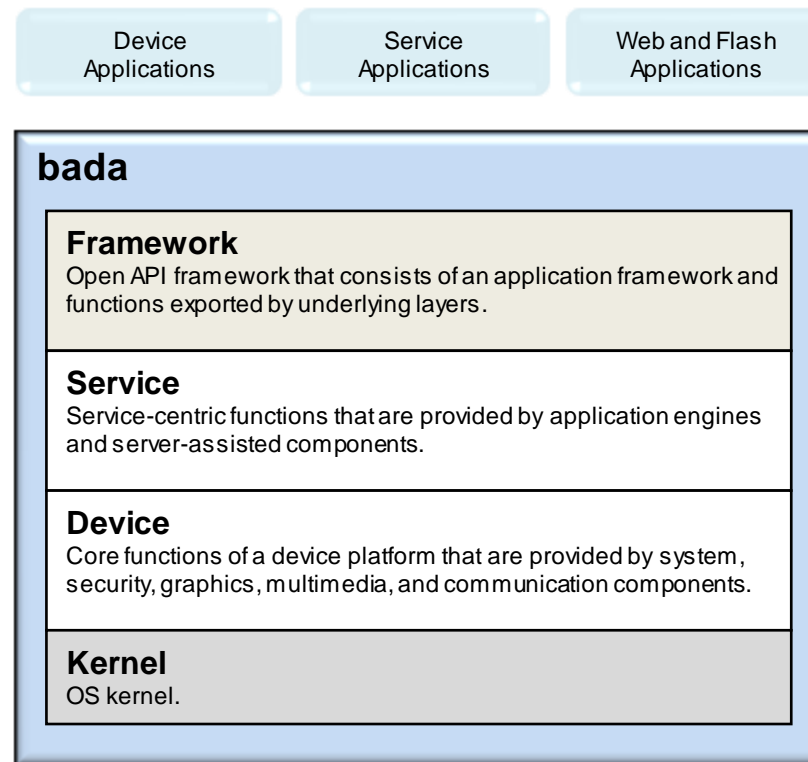
Different service applications can share information such as personal profiles, social relations, schedules, and contents with simple user confirmation, all in order to provide services with greater personal relevance and cooperative service extensions. For example, social networking applications can share user information with commerce or location-aware applications, and share photos from other content publishing applications.

The bada platform is kernel-configurable so that it can run either on the Linux kernel or real-time OS kernels, which makes bada applicable to a wider range of devices than any other mobile operating system. Samsung bada uses Eclipse and a GNU tool-chain, providing developers with a popular and extensible development environment.

What is bada?

bada Architecture

Samsung bada has a four-layer architecture: kernel, device, service and framework layers. The kernel layer can be the Linux kernel or a real-time OS kernel, depending on the hardware configuration. The device layer provides the core functions as a device platform, such as system and security management, graphics and windowing system, data protocols and telephony, as well as audio-visual and multimedia management. The service layer provides service-centric functions that are provided by application engines and web-service components interconnecting with bada Server. Only the framework layer can export the C++ open API. The framework consists of an application framework and functions exported by the underlying layers.



Osp::Base

- Namespaces for basic types, execution environment, and utilities
 - Basic data types
 - Object, String, DateTime, ByteBuffer, Uuid, and other base types
 - Wrapper classes for C++ primitive types, such as Integer, Short, Double, and LongLong
 - Runtime
 - Timer and Thread
 - Synchronization with Mutex, Semaphore, and Monitor
 - Collection
 - ArrayList, HashMap, Stack, and other collection types
 - Object-based collections and template-based collections
 - Utility
 - Math, StringUtil, StringTokenizer, and Uri
 - Standard library support
 - A subset of newlib
 - C++ STL (Standard Template Library)

Osp::Io

- Namespace for input/output data handling
 - File and Directory
 - Basic file and directory operations
 - Data caging per application

Virtual path prefix	Usage	Permission	Delete on boot
/Home	Home directory for an application	r/w	No
/Home/Share	Used to share temporary data with other applications	r/w	Yes
/Share/[appid]	Used to read temporary data of other applications	r	Yes
/Media/Images	Used to read image data	r	No

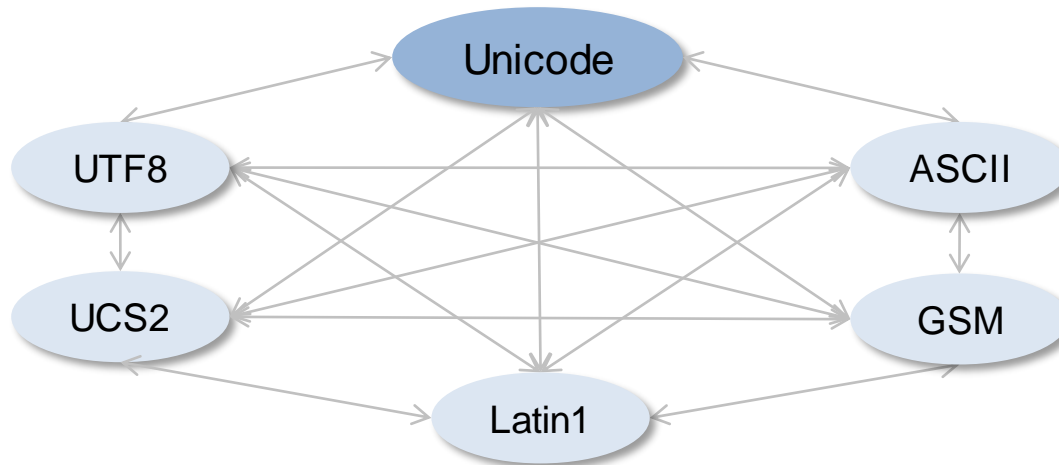
- Database:
 - Transactions such as begin, commit, and rollback
 - Support of both SQL queries and enumerations with DB statements
- Registry:
 - System-provided data store in which applications store and retrieve settings

Osp::Text and Osp::Locales

- Namespaces for internationalization and localization

- Text

- Conversion between the major character encoding schemes, such as UTF8, UCS2, Latin1, GSM and ASCII



- Locale

- Identifying a specific language code, country code, and variant code
 - Formatting locale-sensitive information, such as date, number, and currency
 - Converting a date, time, and number into a string

Osp::App

- Namespace for application framework

- Application types

- UI applications
 - Normal applications with a UI
 - Base applications
 - Limited use only

- Application life-cycle management

- Initializing:

- Resource allocation
 - Loading application states from AppRegistry

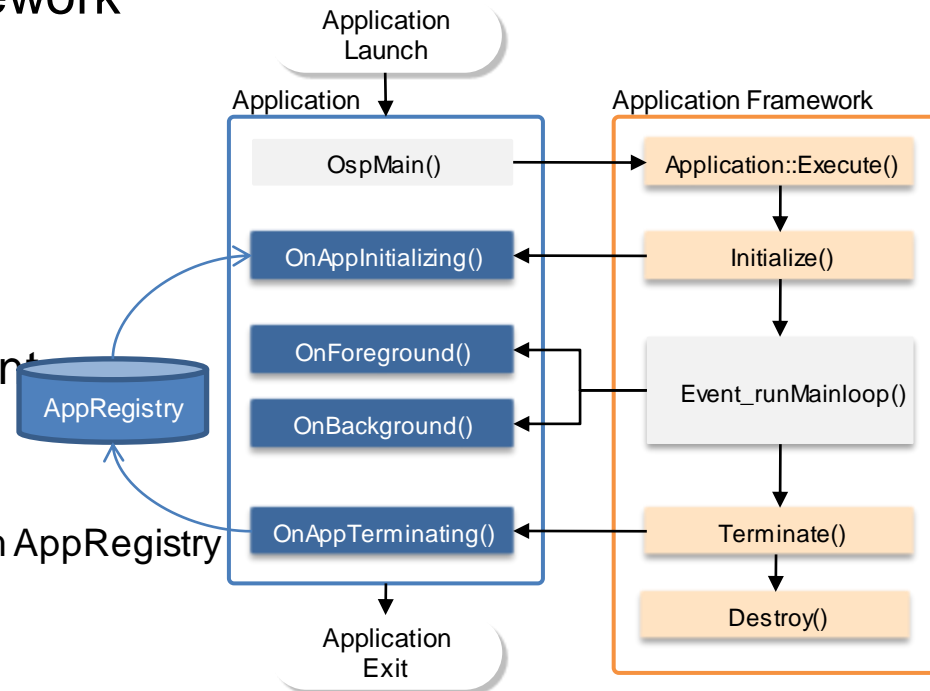
- Terminating:

- Resource de-allocation
 - Providing platform-managed registry to let applications save data
 - System key, memory violation, power off, low memory and battery, and others

- Running:

- UI applications can be sent to the background if an embedded service such as an incoming call is invoked.

- Multiple base applications and one UI application can run concurrently.

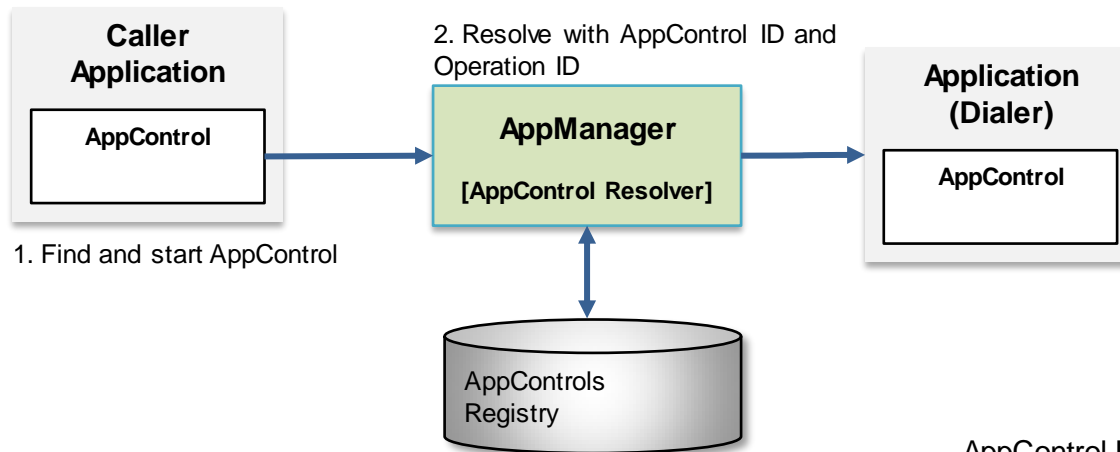


Osp::App

– Application control

- Standard mechanism to consume operations exported by other applications
- Specified with an AppControl ID and an Operation ID:

Application Controls		
APPCONTROL_CALENDAR	APPCONTROL_CONTACT	APPCONTROL_MESSAGE
APPCONTROL_BROWSER	APPCONTROL_EMAIL	APPCONTROL_SIGNIN
APPCONTROL_DIAL	APPCONTROL_CALL	APPCONTROL_MEDIA
APPCONTROL_VIDEO	APPCONTROL_IMAGE	APPCONTROL_AUDIO
...



1. Find and start AppControl

2. Resolve with AppControl ID and Operation ID

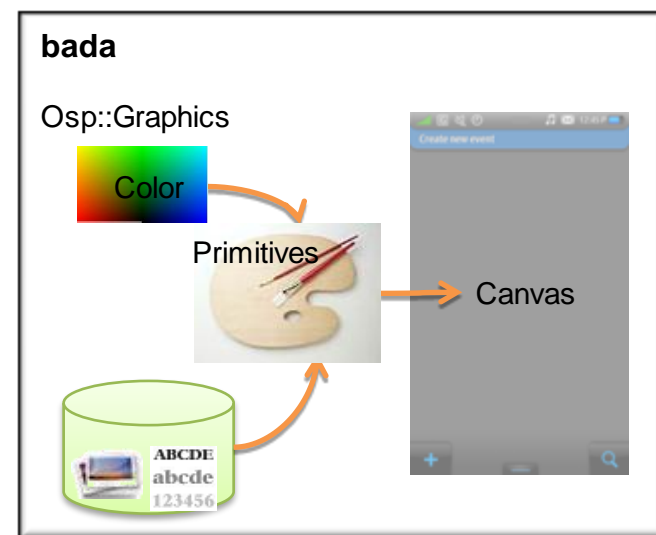


3. Launch the application with resolved AppControl

AppControl ID: APPCONTROL_DIAL
 Operation ID: NONE
 Option: "tel:82312798707"

Osp::Graphics

- Namespace for drawing-related functions
 - 2D graphics
 - 32-bit color (RGB with alpha)
 - Vector fonts
 - System fonts
 - Dynamic font loading
 - Bitmaps
 - 16-bit bitmap with masking color
 - 32-bit bitmap with alpha channel
 - Drawing primitives
 - Point, line, rectangle, ellipse, and polygon
 - Text and bitmap
 - Direct buffer read and write
 - 3D graphics
 - OpenGL® ES 1.1 and 2.0
 - EGL

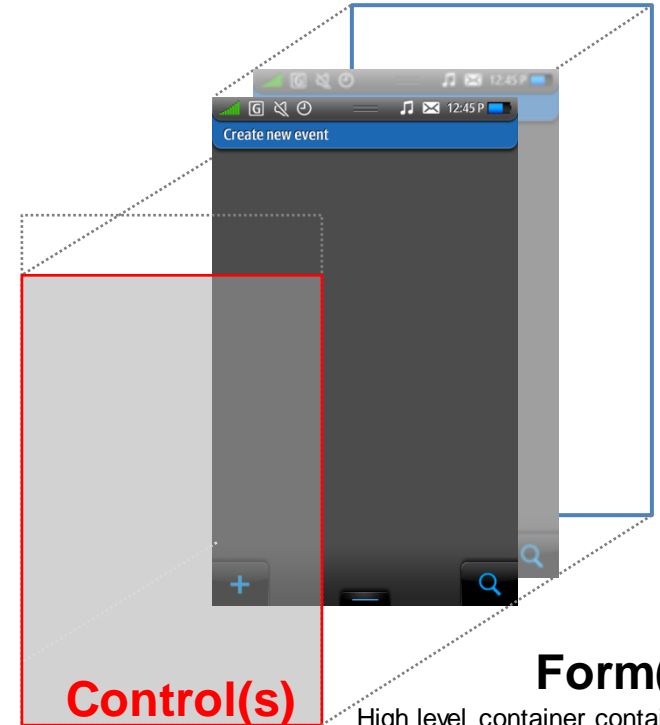


Osp::Ui

- Namespace for creating user interfaces
 - Ui base
 - Event-related classes and interfaces as well as common base classes for all controls
 - Containers
 - Frame, form, popup, panel, menu, and message box
 - Frame and forms support portrait and landscape orientations
 - Controls
 - Common UI controls fully supporting TouchWiz 3.0

Frame

The top level window of an application. There can only be one per application.



Form(s)

High level container containing general controls. There can be multiple forms per application. Each form has an indicator, soft key, and title bar area.

There are multiple user-interface widgets that the application can utilize. These can be placed in any container such as a form.



Osp::Ui

- Advanced controls
 - FlashLite, Animation, Map, and Web controls
- Theme
 - Controls can have themes applied to them.
- Virtual keypad
 - 3x4 keypad, handwriting recognition, and QWERTY keyboards
- XML UI Builder
 - Controls and Containers can be created using the UI Builder. The UI Builder's XML files are parsed at runtime to create the widgets onscreen.
- Resolution independent
 - Controls have the same ratio across different resolutions (WVGA and WQVGA) so that developers can code once to create applications that run on both device resolutions.
- Multi-point touch

Osp::Uix

- Namespace for advanced user interaction
 - Various sensors
 - Acceleration, Tilt, Magnetic, Proximity
 - GPS
 - Weather
 - Programmable vibration
 - Playing vibration effects
 - Magsweep and periodic effects, effect styles, and IVT files from Immersion Corporation
 - Motion
 - Notification for complex and abstract patterns of movement of the device
 - Three types of motions: snap, shake, and double-tap
 - Face
 - Locating & recognizing faces in still images or in camera previews
 - Tracking faces as they move in camera previews



Osp::Net

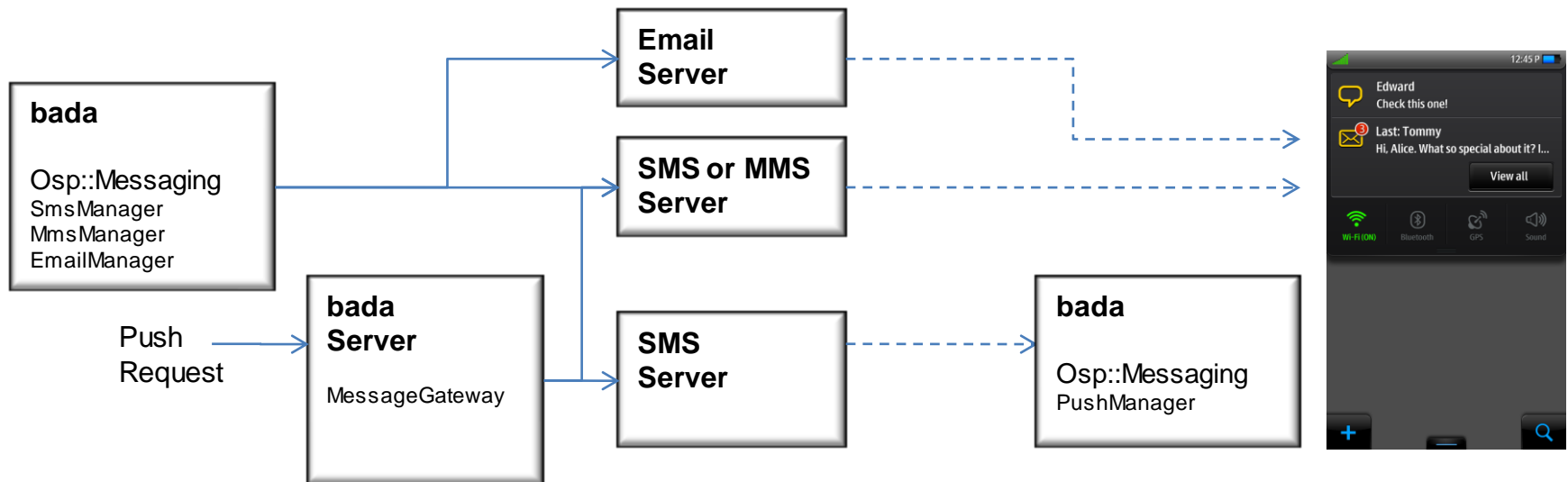
- Namespace for data networking
 - Network management
 - Account management for various bearer information like UMTS and Wi-Fi
 - System default and user-controlled connection management
 - DNS lookup and others
 - Socket
 - TCP client/server, UDP, local socket: non-blocking/blocking mode support
 - TCP client with TLS1.0 and SSL3.0: non-blocking mode support
 - HTTP:
 - HTTP 1.1 including chunking, proxy, and connection management
 - HTTP cookies
 - HTTP basic and digest authentication, and SSL tunneling
 - Bluetooth
 - GAP, OPP, and SPP support
 - Wi-Fi
 - Infrastructure mode and independent mode (ad-hoc service)

Osp::Telephony

- Namespace for telephony capabilities
 - Call
 - Getting current call type and call status
 - Network
 - Getting notifications of network status changes for voice and data services
 - Getting network information, such as PLMN, MCC, MNC, Cell ID, and LAC
 - SIM
 - Getting SIM card information, such as SPN, MCC, MNC, ICC-ID, and operator name

Osp::Messaging

- Namespace for messaging services
 - Messaging service
 - Creating and sending messages: SMS, MMS, and email messages
 - File attachments for specific types
 - Push notification service
 - Receiving push messages from remote servers
 - WAP push depending on service coverage



Osp::Media

- Namespace for multi-media handling
 - Media playback
 - Local audio or video content
 - Streaming audio or video content via RTSP
 - Mixing multiple audio streams
(1 compressed + N uncompressed)
 - Recording
 - Audio and video recording
 - Image processing
 - Encoding, decoding, and conversion
 - JPEG compression
 - Camera
 - Live pictures for previewing
 - Still image captures
 - DRM information:
 - OMA DRM and MS DRM

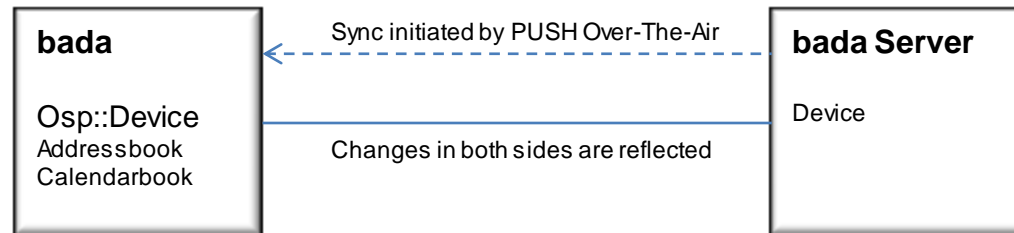
Type	Format	Enc	Dec	File Type
Audio	AAC		O	(.aac)
	AAC+		O	(.3ga) (.m4a)
	Enhanced AAC+		O	
	AMR-NB	O	O	(.amr) (.3ga) (.m4a)
	MP3		O	(.mp3)
	WMA		O	(.wma) (.asf)
	MIDI		O	(.mid) (.spm) (.imy) (.mmf) (.xmf) (.mxmf)
	PCM, MS-ADPCM, IMA-ADPCM, ITU-ADPCM (G.711)		O	(.wav)
Image	JPEG	O	O	(.jpg)
	GIF		O	(.gif)
	PNG	O	O	(.png)
	TIFF		O	(.tiff)
	WBMP		O	(.wbmp)
	BMP	O	O	(.bmp)
Video	H.263	O	O	(.3gp) (.mp4)
	H.264		O	(.3gp) (.mp4)
	MPEG-4	O	O	(.3gp) (.mp4)
	VC-1		O	(.wmv) (.asf)

Osp::Security

- Namespace for cryptography keys, algorithms and certificate management
 - Cryptography keys and PRNG (Pseudo Random Number Generator)
 - Cryptographic algorithms, such as HMAC, Hash, AES, DES, and RSA
 - X.509 certificate information retrieval and certificate path validation

Osp::Device

- Namespace for device sync and management
 - Data sync among user devices
 - Start and stop synchronization process
 - Only one synchronization process is allowed at once.
 - Addressbook and Calendarbook sync
 - vCard 2.1 and vCalendar 1.0 support

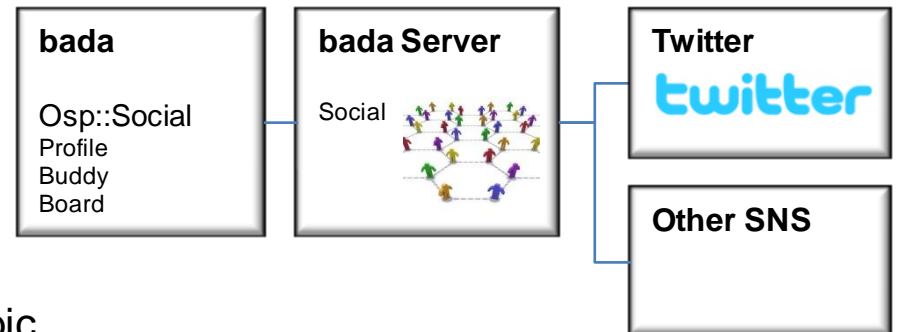


Osp::Social

- Namespace for managing user data stored on a device
 - Addressbook management
 - Local contact DB management
 - Calendarbook management
 - Local schedule DB management
 - Event (appointment and anniversary) and To-Do support
 - Application log query
 - Communication logs

Osp::Social

- Namespace for managing user data in the cloud
 - User profile service
 - Social profile searching and sharing among users
 - Privacy control of exposure level (closed, viewable, searchable) and data accessibility (personal profile, email, phone numbers, and others)
 - Buddy management
 - Buddy requests between users
 - Accessing user's buddy list
 - Board service
 - Topic creation and search
 - Exchanging comments on a topic
 - Messaging service
 - Exchanging text messages between users within applications
 - SNS gateway
 - Unified interface to access various social networking services, such as Twitter

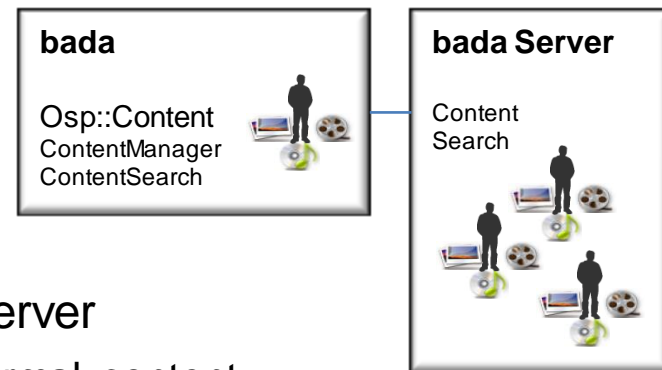


Osp::Content

- Namespace for content management and search services on a device
 - Local content management
 - Lifecycle management
 - Content information
 - User tagging info, such as comments
 - Media metadata, such as ID3 and EXIF
 - Context info, such as GPS information
 - Local content search
 - List of values for specific columns
 - SQL-like search

Osp::Content

- Namespace for content management and search services in the cloud
 - Content management on server
 - Life-cycle management
 - Content information
 - Various user-defined content types, such as image, video, audio, game, and text
 - Various user-selected or defined attributes (metadata)
 - Content search on server
 - Content type and category list
 - Conditional search
 - Keyword search
 - Popular or related terms
 - Content transfer to and from the server
 - HTTP upload and download for normal content
 - OMA download for DRM contents



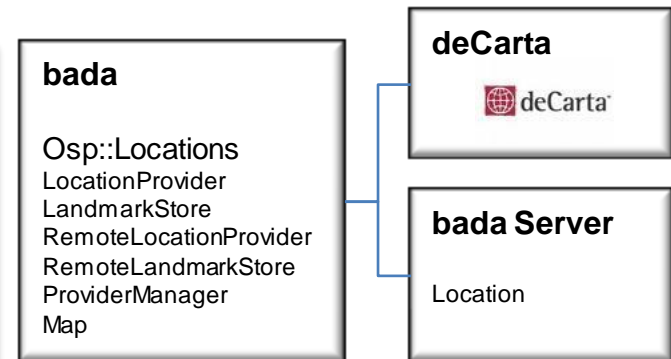
Osp::Locations

- Namespace for location-related information and services
 - Locating
 - GPS
 - Location updates and retrieval
 - Periodic location updates
 - Getting the locations of other devices, such as last known or current location
 - Subscribing to zone and trace services
 - Privacy protection
 - Landmark management
 - Landmark store management
 - Create, read, and delete
 - Device-wide or application bounded
 - User-created landmark management
 - Add, update, delete, and search
 - Pre-defined landmark search



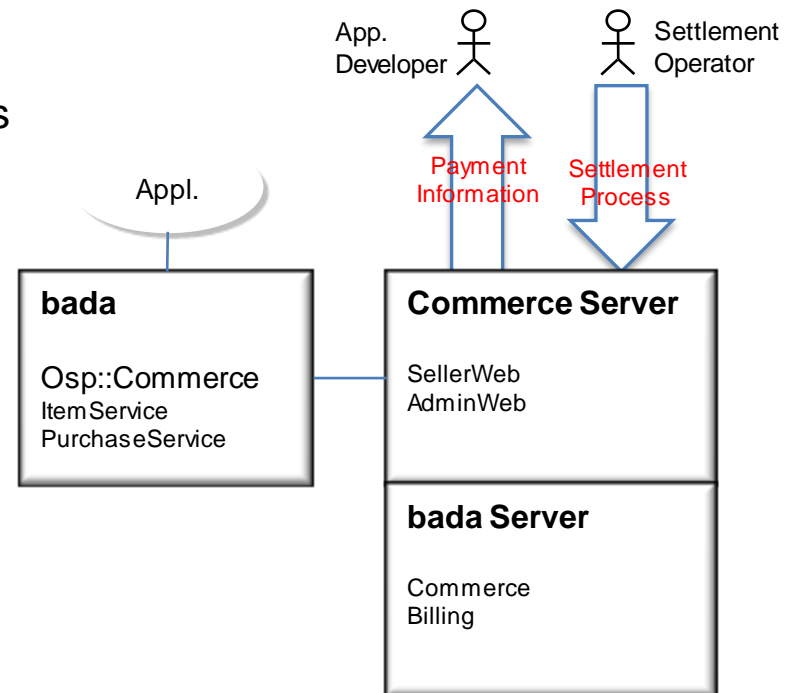
Osp::Locations

- Location services from deCarta
 - Map rendering, zooming, and panning
 - Geocoding and reverse geocoding
 - Directory services
 - Route services
- Map UI control
 - Map zooming and panning
 - Overlay and information windows
 - Event notifications



Osp::Commerce

- Namespace for commerce services through the Samsung Application Store
 - Item information queries
 - Getting item information in applications
 - Managing item information, such as item name and description, through the Samsung Application Store seller office
 - Item purchases
 - Purchasing items in applications



Osp::Web

- Namespace for embedded browsing functionality
 - Web control
 - Rendering web contents in a Web UI control without running a browser application
 - Browser utilities
 - History management
 - Download functionality
 - Accessing URLs and fetching data
 - Intercepting URL requests and giving an application a chance to handle the request
 - Routing fetched data to applications
 - JavaScript utility
 - Evaluating JavaScript code given by an application and returning the result of the code





The platform with more opportunities
Invitation to Adventure