Blee and BxGnome: ByStar Software-Service Continuum Based Convivial User Environments

Document #PLPC-180004
Version 0.1
September 24, 2012

This Document is Available on-line at:
http://www.persoarabic.org/PLPC/180004

Banan, et al.
Email: http://www.persoarabic.org/contact

Contents

List of Figures

Part I

The Model Of ByStar Software-Services Continuum

This document is about "The ByStar User Environments Model" which is part of:

The Libre-Halaal ByStar Digital Ecosystem
A Unified and Non-Proprietary Model For Autonomous Internet Services
A Moral Alternative To The Proprietary American Digital Ecosystem

With ByStar, we are creating a complete parallel digital ecosystem to compete with and stand against the existing Proprietary American Digital Ecosystem. Above all, ByStar is about preservation of the individual's autonomy and privacy.

1 The ByStar User Environments Model

Figure 1 illustrates how ByStar User Environments fit in the overall ByStar digital ecosystem.

There are two types of ByStar User Environments:

- ByStar Libre-halaal Emacs user Environment (Blee)
- ByStar Gnome and Browser User Environment (BxGnome)

ByStar user environments are available on the following form factors:

Supported Form Factors: Desktop, Laptop, Netbook, Handset, Tablets

ByStar user environments are tightly coupled with ByStar Autonomous and Controlled Libre-Halaal Services including.

ByStar Autonomous and Controlled Libre-Halaal Service: ByName, ByFamily, ByAlias, ByMemory, BySmb, ByWhere

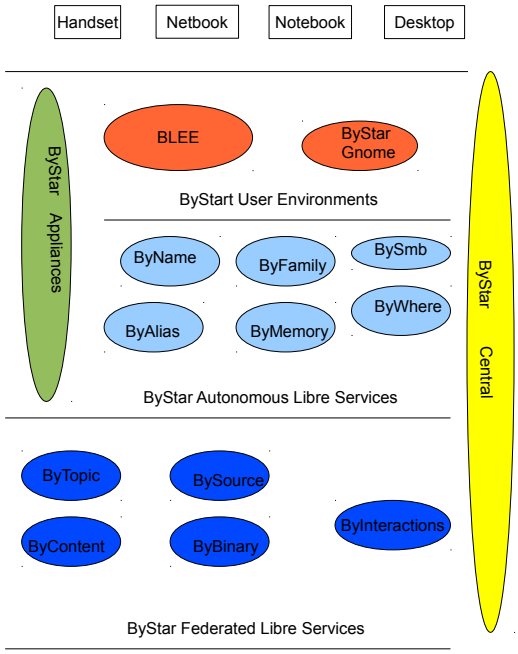

Figure 1: ByStart Entities

2 The ByStar Autonomous Libre Services Model

This documents Autonomous Libre Services and access to the services through the browser.

Usage through BUE is described in xxx.

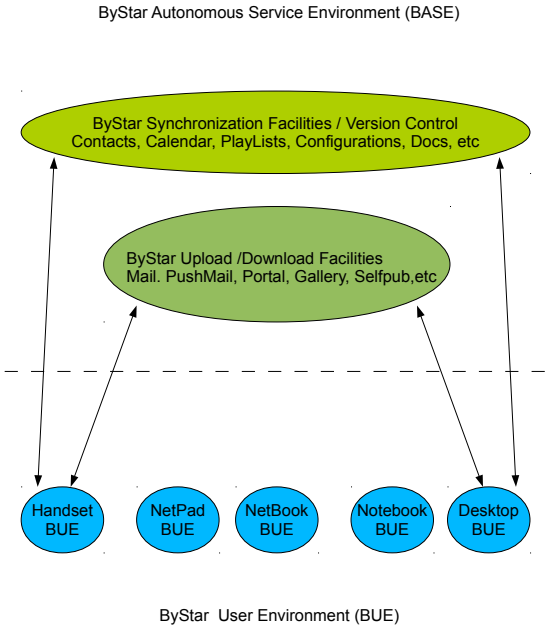

Figure 2: Interface Model of ByStar User Env and ByStar Services

3 Structure Of ByStar User Environments

3.1 ByStar Libre-halaal Emacs Environment (Blee)

Emacs is far more than just an editor. Viewing Emacs as an Editor Centered User Environment is only the beginning of recognizing its power. Beyond a user environment, we have been using Emacs as Software-Service Integration Framework for many years. We are now in the process of packaging the entire environment as the ultimate User Environment For the Software-Service Continum. This package we call BLEE. The By* Libre Emacs Environment. Blee goes beyond Emacs by fully integrating Emacs and Firefox/Iceweasel and all of Linux apps underneath it. BLEE features include: – Emacs <—-> Firefox (two way integration) – Inside of Firefox: edit everything through Emacs, Send emails using Emacs+Gnus, ... – Inside of Emacs: write html and view it in native firefox in realtime, View all your Gnus email messages in full html in firefox, ... Think of Emacs and Firefox as joint sisters. They are all you need. – Incredible Email capabilities Built on top of Gnus. Awsome Multi-Mailbox and Multi-Address support provided with ByStar. Great Anti-Spam capabilities, Search Capabilities, Scoring based on full integration with the address book. Mailing lists access through News with Gmane. – Organizational Tools the likes of which exists no where else: Address Book: bbdb (Big Brother DataBase), Calendar/Diary, Org Mode: To Do Lists, Worklogs, Dynamic Blocks, – Full Integration With dict (Multi-Lingual Dictionaries, Thesarus, ...), Powerfull templating systems and Abbreviations. – Fully Multi-Lingual (m17n). Emacs 24 now includes bidi (bidirectional editing). That means in addition to Latin being native, now Perso-Arabic script is also native emacs. – Of course, Music (emms) and tons of games. And all of this we will show you. This talk will for the most part be interactive demos.

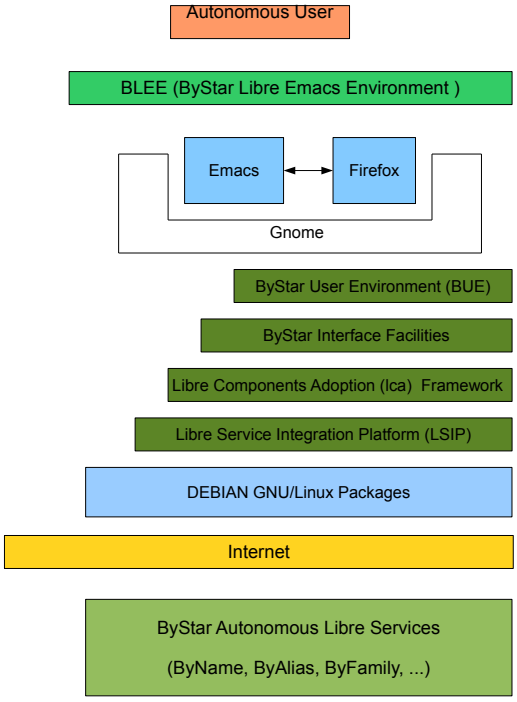

Figure 3: ByStar Libre Emacs Environment (BLEE) Model

4 ByStarEntity Model And Terminology

4.1 The ByStarEntity Concept

By* is based on a set of key abstractions, representing the major real-world entities that must be represented within a generalized web structure. These entities include such things as individual persons, businesses, physical locations, and events. For each such entity we have defined the structures and conventions required to represent, instantiate and name that entity in a unified, consistent way, and at a very large scale. We have then defined the major classes of services required to manage these entities, and to allow highly generalized interactions within and among each other.

In the ByStar applied model, a real-world entity type (for example individuals or a physical locations) maps on to a `ByStarEntityType` (`BxEntityType`). A real-world entity instance maps on to a `ByStarEntity` (`BxEntity`) All ByStar services are anchored in `ByStarEntity`.

ByStarEntityTypes are structured hierarchically in a tree.

`ByStarEntityType` is either a `ByStarAutonomousEntityType` or a `ByStarControlledEntityType`.

`ByStarAutonomousEntityType` and `ByStarControlledEntityType` are either Classified or UnClassified.

In this structure, persons identified by their name, are represented as:

```
ByStarEntityType=ByStarAutonomousEntityType.Classified.Person.ByName
```

Each `ByStarEntity` (an instance) is identified by `ByStarEntityId`.

A ByStarEntityId is structured as:

```
ByStarEntityId=RegistrarId+ByStarEntityType+InstanceId
```

All ByStarEntityIds are unique. The InstanceId is assigned by the RegistrarId.

Each ByStarEntity can be activated within a ByStarAutonomyAssertionVirtualMachine (BxAutonomyAssertionVirtualMachi
The representation of a ByStarEntity in a ByStarAutonomyAssertionVirtualMachine
is called a ByStarServiceObject.
A ByStarServiceObject maps to a Unix account and a user-id.
The ByStarServiceObject can have any ByStarServiceCapability
that ByStarAutonomyAssertionVirtualMachine offers.

Any ByStarServiceCapability can be bound to and exposed through a registered domain name.

Currently, ByStarServiceCapability is one of the capabilities enumerated in figure **??**.

Based on the above structures, ByStar services can consistently grow and interact with other ByStar services to provide a rich and healthy environment.

4.2 Key Concepts

- ByStar Account (sa-20000)
- ByStar Account Fully Qualified Mail Address
- ByStar Account Fully Qualified Domain Name

4.3 Terminology

- BARBD – ByStar Account Requested Base Domain – Passed to BARC
- BAABD – ByStar Account Assigned Base Domain – Passed to BARC
- Primary, Secondary, Alt, ...

Part II

ByStar Capabilities Overview

Part III

ByStar Email Facilities

5 ByStar Email Model

5.1 Relevant Literature

- End-to-end Arguments in System Design http://web.mit.edu/Saltzer/www/publications/endtoend/endtoend.pdf

[2] are also included in the References list in article format.

5.2 Multi-Address Multi-Mailbox Paradigm

- Use of purposeful addresses in the from line
- ...

6 Squerrelmail: ByStar Web Based EMail Environment

6.1 Email Origination

7 Send Mail

Introduction to Sending Mail. Weave in with the menus

8 Reading and Processing Mail

Introduction to Reading and Processing Mail. Weave in with the menus.

8.1 Gnus Level Facilities

```
l    List all groups that have unread articles.
     With numeric prefix and lower.

L    List all groups With numeric prefix and lower.

A l
    List all unread groups on a specific level.
    With a prefix, also list the groups with no unread articles.
```

8.2 ByStar Level Conventions

```
0- unused

1- Urgent:  text, urgent, mobile,

2- Important / Noraml / Usual

3: Confirmation/Exceptions  /Spam (attention)

4: Reports/FYI/ Self Originated: Archives / References / Bookmarks /Sent /Record --

5: Lists Primary
```

8.3 Address to Final Delivery Mail Box (FDMB) Mapping

8.3.1 Address and Fdmb Purposes

```
* Syntax
========
  - address@
  - [fdmb]  -- no spam protection
  - (fdmb)  -- spam protected
  - <level> --

** fdmb to address mapping and purposes
---------------------------------------

[.]       main@
           Purpose:

[test]    test@
           Purpose:

[text]    text@
           Purpose:

[school]    kid@ kids@ shs@ tillicum@
           Purpose:
```

Part IV
ByStar Web Facilities

9 Plone 3

9.1 Relevant Literature

- Plone http://www.plone.org

[1], [3] are also included in the References list in article format.

9.2 Interactive Plone

Figure **??** shows ...

Part V

ByStar Content Development and Content Publication Facilities (LCNT)

10 Blee Self Publication Facilities

Introduction to Blee Self Publication Facilities. Weave in with the menus.

10.1 Creating A New Document

```
- lcnLcntGens.sh -n showRun -p cntntRawHome=. -e "Start Blank" -i lcntBaseStart ttytex main

- Assign a number
echo /lcnt/lgpc/bystar/permanent/usage/blee/main.ttytex >> /lcnt/lgpc/bystar/SOURCE-INFO/permanent.reg
Then assign a number in: /lcnt/lgpc/bystar/SOURCE-INFO/permanent.reg

- lcnLcntGens.sh -p cntntRawHome=. -i lcntRefresh

--   This generates the LCNT directory with
      default content.
--  Creates lcntProc.sh

- Then customize the ./LCNT-INFO Directory

  cd LCNT-INFO

  At a minimum edit:
        mainTitle
        subTitle
        subSubTitle
        shortTitle
        description

- lcnLcntGens.sh -p cntntRawHome=. -i lcntBaseConfig article

- Edit the document -- Run dblock

- Run ./lcntProc.sh
```

10.2 Creating A New Document

10.3 Creating A New Document

Part VI
ByStar Photo Gallery

11 Gallery Upload

11.1 Gallery Folder Upload

- ftp from Droid

Part VII
ByStar Genealogy

12 Geneweb

12.1 Gallery Folder Upload

- ftp from Droid

Part VIII
ByStar Multimedia (Audio/Music Video) Facilities

13 Music

13.1 Overview

- Bystar Music as a Web Service – Described Here
- Bystar Music as part of User Envirenvironment – Described in PLPC-180002

At this time, the entire music related information is maintained in PLPC-180002.

Part IX

Maintenance and Development Of ByStar User Environments

14 Bugs and Todos

Bugs and Todos go here.

15 Ideas

Ideas will be captured here.

And here we go.

References

[1] Andrew Hammoude ” ” Mohsen BANAN. ” lessons from history: Comparitive case studies ”. Permanent Libre Published Content ”100017”, Autonomously Self-Published, ”August” 2000. http://www.freeprotocols.org/PLPC/100017.

[2] J. Kempf, R. Austein, and IAB. The Rise of the Middle and the Future of End-to-End: Reflections on the Evolution of the Internet Architecture. RFC 3724 (Informational), March 2004.

[3] D. Thaler and B. Aboba. What Makes For a Successful Protocol? RFC 5218 (Informational), July 2008.