

Leibniz-Institut für Wissensmedien

SambaXP 2015



Introducing Bebop to Samba 4

Torsten Kurbad

Leibniz-Institut für Wissensmedien Tübingen t.kurbad@iwm-tuebingen.de

Outline

Introduction

Development of Our Samba Domain

A Short History of Bebop and its Predecessor(s)

Why Manage AD Users and Groups with Bebop?

Live Demo

Other Use Cases of Bebop

Conclusion



Outline

Introduction



Bebop as a Musical Style

Bebop or **bop** is a style of jazz characterized by a *fast tempo*, instrumental virtuosity and *improvisation* based on the *combination* of harmonic *structure* and sometimes *references* to the melody. [...] This style of jazz ultimately became synonymous with *modern jazz* [...]

Source: Wikipedia

How does that definition translate to a software?



About my Employer

- Leibniz-Institut für Wissensmedien / Knowledge Media Research Center
- Non-profit multidisciplinary extra-faculty research institute situated in Tübingen, Germany
- Main field of research: Utilization of digital media in teaching and learning
- Founded in 2001 with about 30 employees, most of them (cognitive) psychologists.
- 2014: 185 employees, i.p. 81 scientists of multiple disciplines, 38 people in service areas,
 66 student assistents



About my Employer

Student assistents...

... have very short-lived work contracts

... are encouraged to try out different fields of research, thus do frequent work group hopping

→ High fluctuation poses challenges for both HR and IT



About me

- Computer scientist
- First contact with Linux in 1995
- Joined the IWM in January 2004
- Hired as Python programmer, now almost full time server administrator
- Advocate of open source solutions



Outline

Development of Our Samba Domain



January 2004 - Situation

 Windows NT style domain with roaming profiles on the Linux DC based on Samba 2

 Disjunct user databases for almost every machine/service



January 2004 - Assessment

- Substantially outdated Samba version
- → Various username / password combinations per user
- → No unified user or group management



End of 2004 – Situation Changes

- Windows NT style domain based on Samba 3,
 OpenLDAP, and MIT Krb5
- POSIX groups in OpenLDAP
- Unified Windows / Linux authentication utilizing Samba, pam_ldap, nss_ldap, and pam_krb5
- Several services still utilizing isolated user databases
- Samba user management with smb-ldap-tools



End of 2004 - Assessment

- → Up-to-date Samba version
- → Fewer username / password combinations per user
- Partly unified user / group management,
 not usable by laymen*

*layman pl.: laymen = non-geek



2008 – Situation Changes

Some (non LDAP-aware) services still utilizing isolated user databases

Samba user / Posix group management with Bebop



2008 - Assessment

→ Samba 3 still up-to-date, but growing demand for AD

Mostly unified user / group management, usable by laymen



Summer 2014 – Situation Changes

- NT style domain upgraded to Active Directory based on Samba 4.0.18
- Unified Windows /Linux authentication based on Samba AD, nslcd, and pam_krb5
- Most services utilizing AD user / group database
- But: Samba AD user / group management with samba-tool and bash scripts



Summer 2014 - Assessment

→ Up-to-date Samba version

Mostly unified user / group management, again not usable by laymen



2015 – Situation Changes

Active Directory domain upgraded to Samba 4.1.x

 RODC in DMZ, based on Samba 4.1.x, to enable domain authentication for DMZ / external services

AD user / group management with Bebop



2015 - Assessment

→ Up-to-date Samba version

 Unified user / group management, usable by laymen



Future Plans

Let all services utilize AD user / group database!

Exactly one username / password per user (SSO?)



Outline

A Short History of Bebop and its Predecessor(s)



2002-2007 - VisualGroup

Group-aware, document-centered collaboration utility

- Written in Python2.1, based on the Zope2 web framework
- Presented documents in a folder view
- Supported user logins, ACLs, and versioning of all content, but user database entirely isolated
- All content was being held in the Zope object database
- All operations done server side, i.e. no Javascript
- Used to store publications and internal documents
- Sorry, no screenshot survived :-(



2002-2007 - VisualGroup

Pros	Cons
+ Intuitive web interface to manage documents	- Intuition = programmer's intention, difficult to apply to a large group of people
+ Transparent versioning of all content	- With more and more content, the user interface became increasingly slower
+ Transactional operations	 Internal user database and login scheme → limited interconnection possibilities
+ Wide variety of supported platforms/browsers	- Based on obsolete versions of Python and Zope

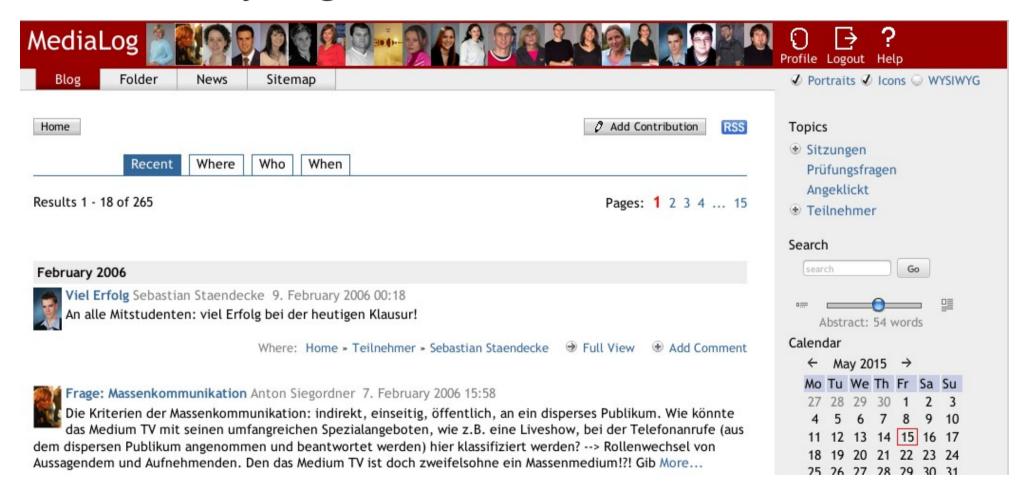
2006-Today - Bebop 1.0

Group-aware, collaboration-centered document and content management utility

- Written in Python2.4, based on an early development snapshot of the Zope3 framework
- Supports different views on the same content,
 e.g. Wiki, Blog, folder structure → "content improvisation"
- AJAX interface including WYSIWYG editor
- Highly configurable → most features can be enabled or disabled on demand
- All content is being held in the Zope object database
- Supports LDAP user authentication, recently adapted to Active Directory

2006-Today – Bebop 1.0

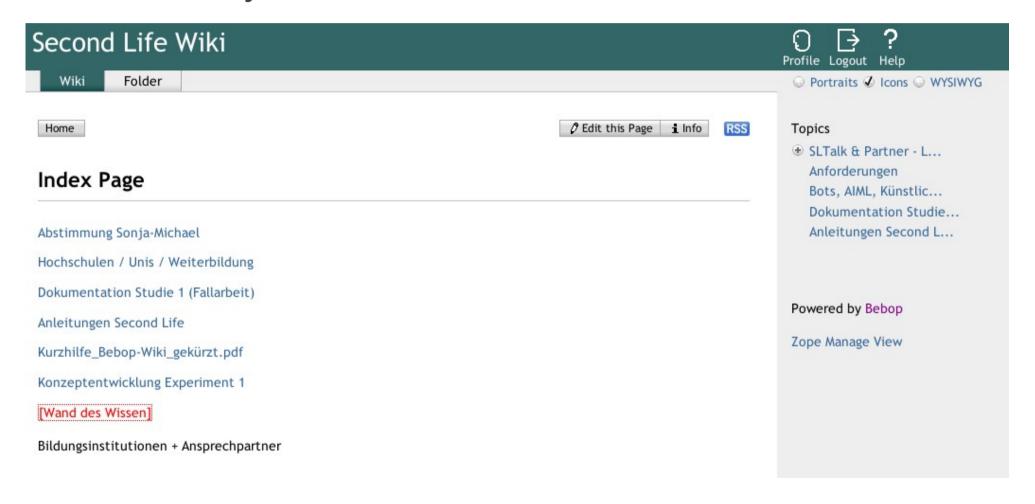
Screenshot of Blog view





2006-Today – Bebop 1.0

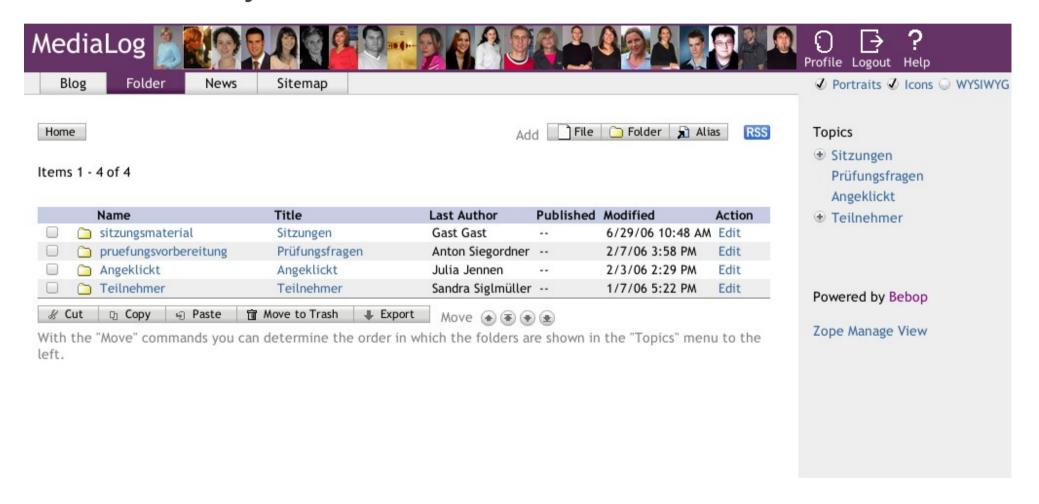
Screenshot of Wiki view





2006-Today – Bebop 1.0

Screenshot of Folder view





2006-Today – Bebop 1.0

Pros	Cons
+ Flexible web interface, based on user feedback and demands	- Flexibility = complexity We noticed that most users like it plain and simple (or began using Google docs)
+ Client side technologies for faster response times	- Content stored in object database → large and slow
+ Still very popular with our scientists	 Based on development snapshots and obsolete Python version → difficult to maintain
+ Allows LDAP / AD authentication	- Does not allow manipulation of LDAP / AD objects

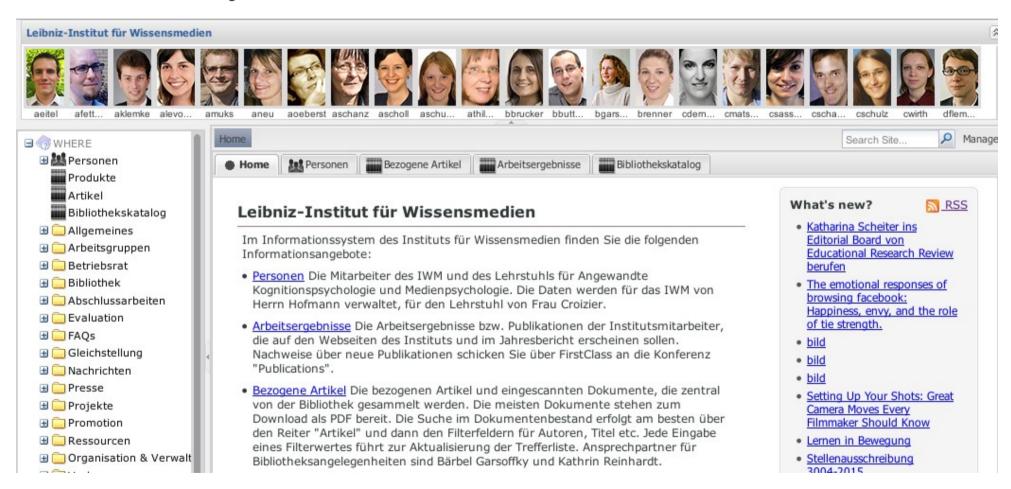
Late 2007-Today – Bebop 2.0

Group-aware, collaboration-centered document and content management utility

- Written in Python2.7, based on release 3.5 of the Zope3 framework
- **Combines** different views on the same content, based on the questions "where", "when", and "who"
- Responsive client side ExtJS interface
- Highly configurable
- Easily extendable by plugins
- Large objects stored in the filesystem
- Supports LDAP user authentication and (since 2008)
 manipulation, recently adapted to Active Directory

Late 2007-Today – Bebop 2.0

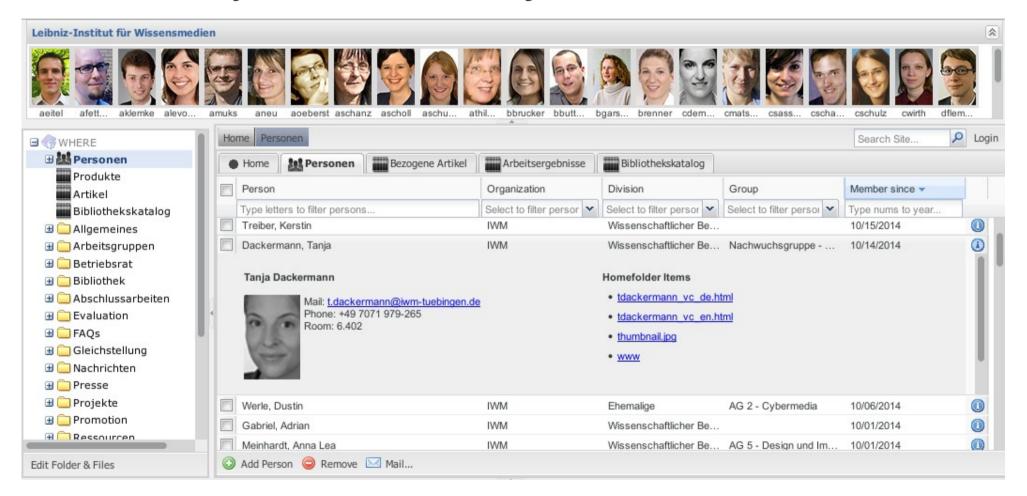
Screenshot of Main Window





Late 2007-Today – Bebop 2.0

Screenshot of Persons Directory





Late 2007-Today – Bebop 2.0

Pros	Cons
+ Web interface based on combination of folders and logical / "contextual" views "where", "when", "who"	- Based on now obsolete versions of Python and Zope3 → will soon become difficult to maintain
+ Large content objects stored in filesystem, database holds references only	- Not always very responsive → <i>faster</i> speed probably achievable by optimization
+ Highly configurable in many areas	- Larger customizations require Python programming skills
+ Supports LDAP / AD auth and manipulation	- Has to keep the Zope database in sync with the AD



Outline

Why Manage AD Users and Groups with Bebop?



Why Manage Your AD with Bebop?

Because...

- ... it's **easy**
- ... it's **open source**
- ... it's **flexible**, without being too complex
- ... it helps ensure proper **formatting** of AD fields
- ... it can help reduce redundancies
- ... it **reduces** the **workload** of both HR and IT
- ... it **reduces** type **errors**
- ... it provides you with a nice **web**-based **view** of your user / employee database

• • •



Why Manage AD with Bebop?

Because...

→ Allow me to **demonstrate**



Outline

Live Demo



Outline

Other Use Cases of Bebop



Other Use Cases of Bebop

Examples

- Collection and management of publications and other intellectual products
- Management of libraries
- Management and display of website(s)
- Management of (technical) resources
- With all of the above in one place, accessible through a single interface, exportable in open standard formats



Outline

Conclusion



Current State of Affairs

- Bebop 2.0 in active use for more than 7 years
- Integral part of our workflows
- Accepted by both HR and IT to manage employee / user data
- But: Code base outdated, hard to maintain
- But: Several shortcomings and errors surfacing in daily use
- → Would be nice to have a **successor**...



(Uncertain) Future Goals – Bebop 3.0?

- Upgrade code base to Python 3.x, recent versions of Zope3 (now BlueBream) and ExtJS (now sencha)
- Improve code optimization for faster response times
- Make it easier to customize
- Use external data sources (like AD) directly instead of synchronizing them to the Zope database
- Improve testing and test coverage
- → Unfortunately (too?) much effort for three programmers



In Case You Want to Try out Bebop Yourself

It's easy

Check it out via subversion:

svn co http://svn.kmrc.de/projects/buildouts/bebop.minimal

Follow **README_AD.txt** to create a Bebop site and connect your Active Directory

For questions and suggestions, get in touch:

mailto:t.kurbad@iwm-tuebingen.de



Thank **you** for your attention!

