Dev.Design Issues

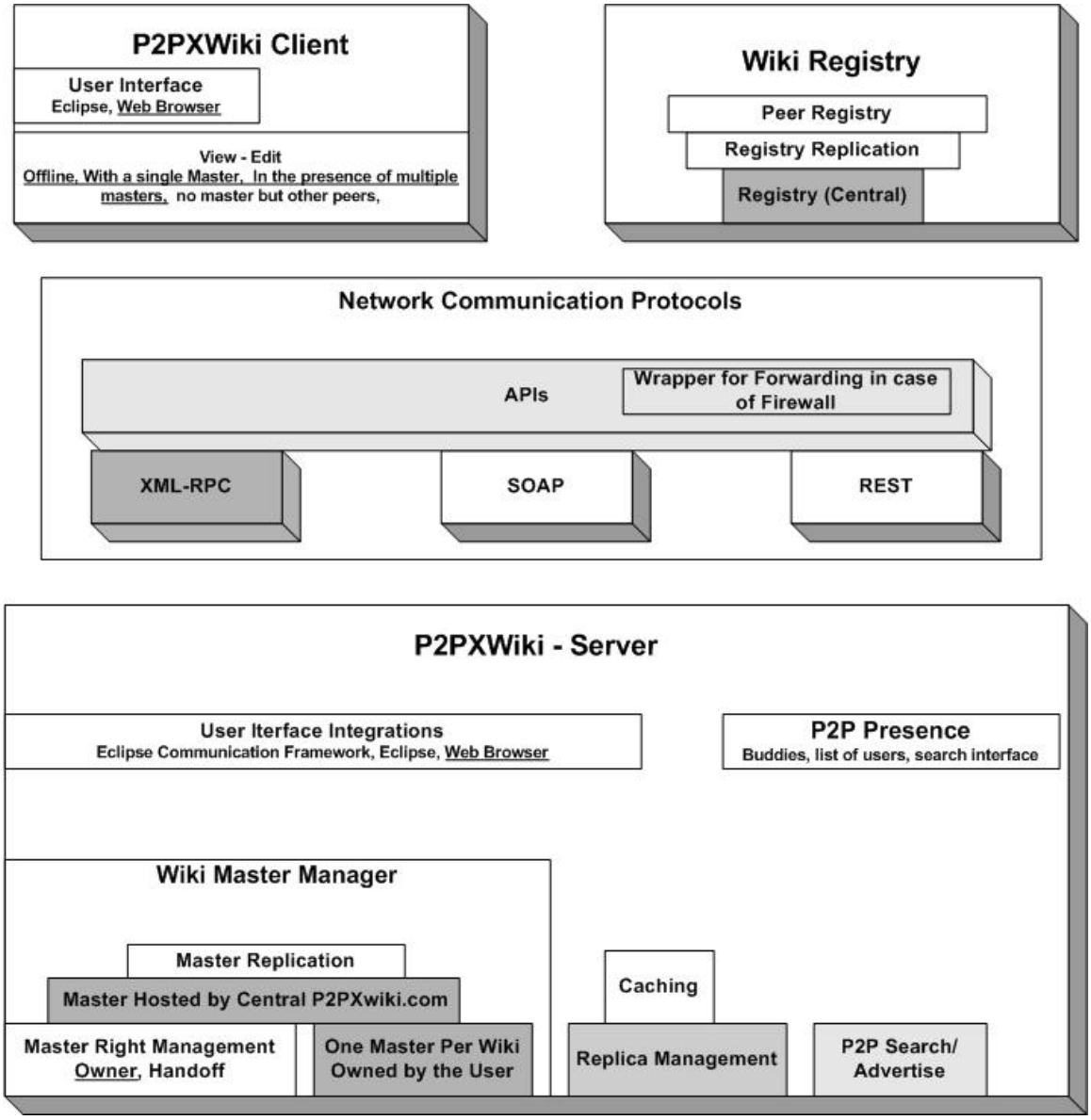
Design Questions, Choices, and Decisions

The Discussion has been copied over here.

Design Decisions

- A master-slave architecture, where the creator of the wiki is the master. Others are slaves. Master is on a wiki by wiki basis as opposed to pages or all the wikis. So a sinlge wiki is the granularity for replication/consistency.
- The master need to be present for the edits on that particular wiki by other peers.
- - extension: allow a peer to access a wiki when the master is offline by handing off master rights
- There is a central registry for name resolution. The central registry allows for the wiki names to be distinguishable. Also it helps in solving the problem of wiki-hijacking. More on this is in the discussion at the link above.
- - extension: central registry can be replicated between a set of servers
  - extension: central registry functionality can be handed by peers. There will be a need for ensuring the integrity of the registry data in this case.
- The registry need to be present before a new wiki can be created.
- Allow a user to access p2pxwiki without the p2pxwiki software by contacting a hosting service. This helps in having a wiki at a public place accessible by everyone with a browser.
- Allow a user to create a new wiki by downloading the software p2pxwiki. This provides some motivation for users to dedicate space for replicating other people's wiki. For only accessing other people's wiki, user need not download the software.
- Allow the user to get xwiki from other users for read only when the user is not connected to the master or the internet. The replicas at other peers are for a complete wiki and are *full read-only replicas*.
- - extension: Allow a user to write to a xwiki while offline
- Extension: Allow two offline users to share a wiki in read/write mode.

Technical Components



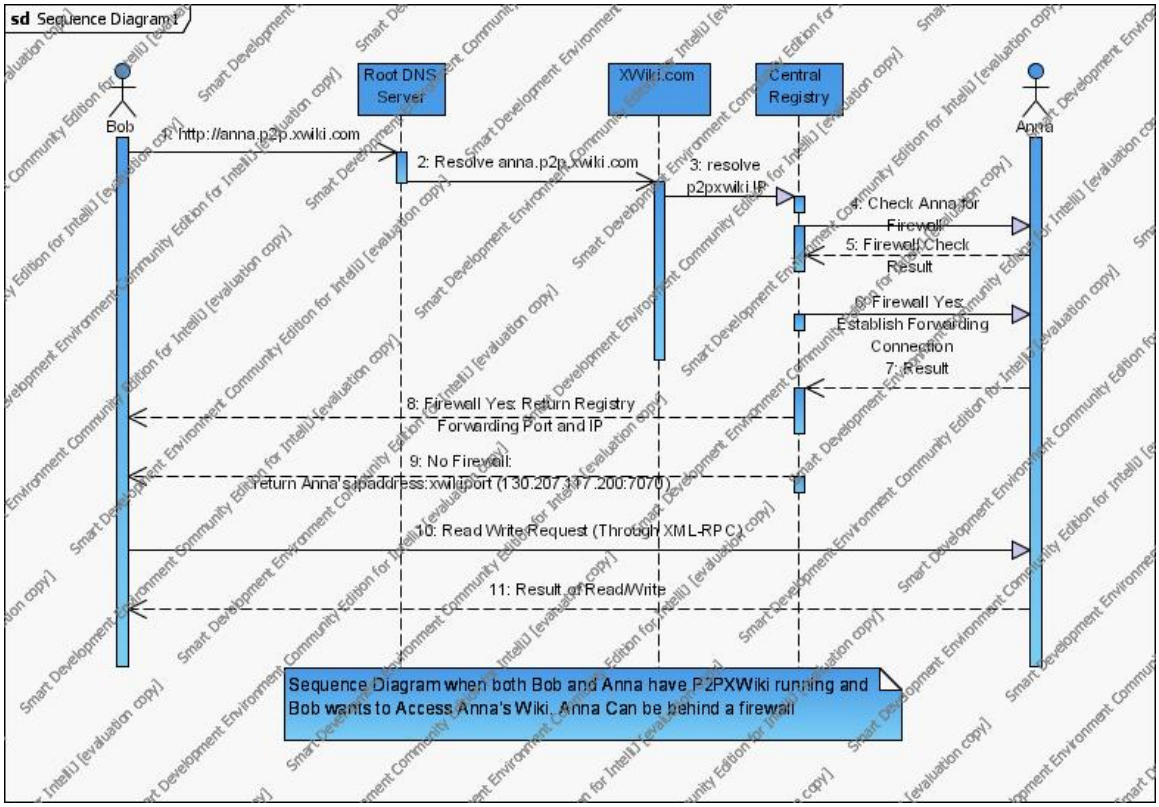
Road Map

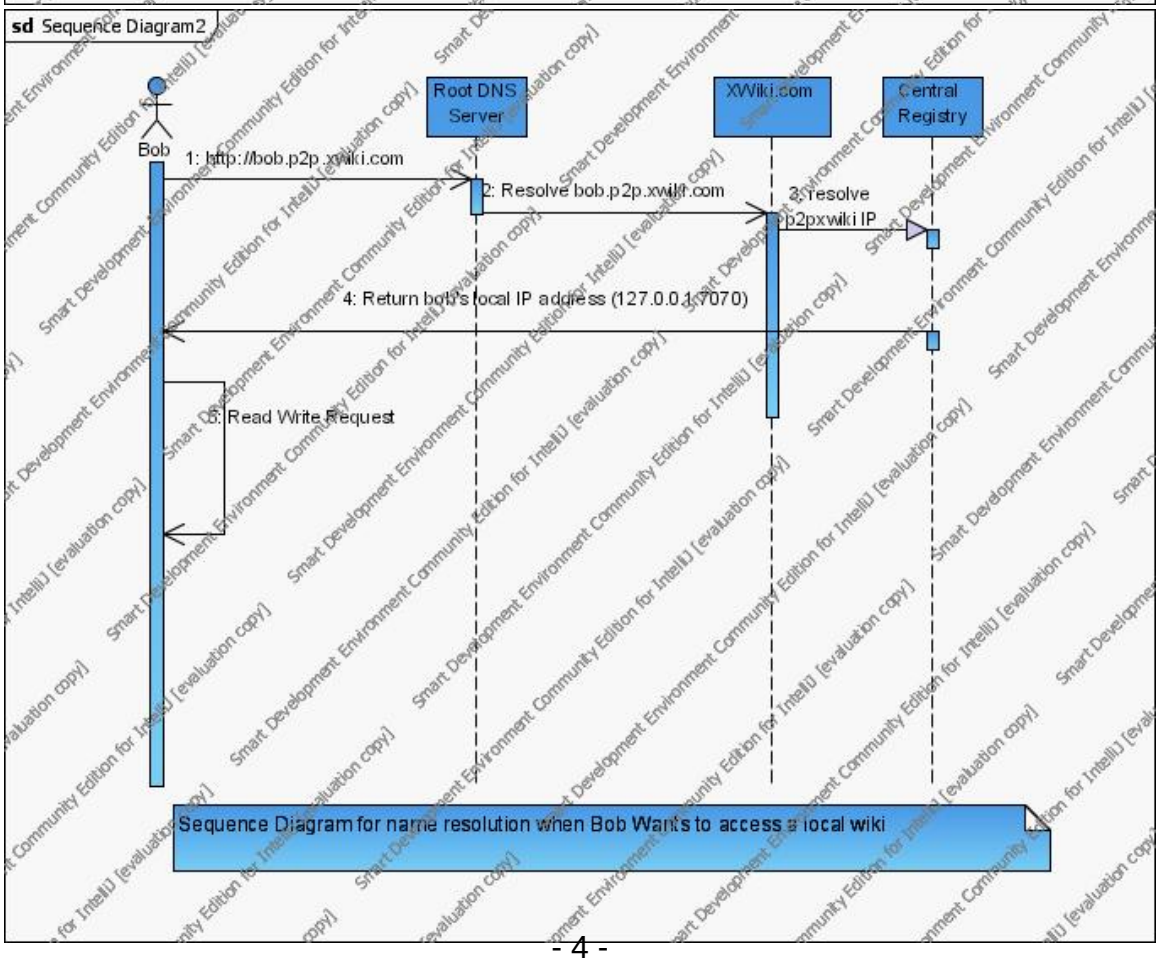
- **Release 0.1: Allows peer to access wiki from each other while everyone is online**.
- ◦ include Name resolution
  - ◦ central registry with possibly a DNS server.
  - ◦ XML-RPC changes to handle virtual wiki mapping, all read/writes will go to the wiki master after name resolution, new apis to handle complete wiki transport.
  - ◦ Allow peers to edit using local copy, however, whenever a peer saves the document, it will go to the master.
- **Release 0.2:Allow Peers to access wiki from a central server when the master is offline**
- ◦ replicate the wiki to a central hosting service.
  - ◦ Extensions to XML-RPC protocol
  - ◦ · A full replication API (get a list of documents since date XXX, multiple updates to documents, etc...)
  - ◦ p2p discovery to find out the location of a wiki
  - ◦ allow read/writes while the Master or the Central wiki is online
  - ◦ provide consistency between the master and central wiki so that wiki semantics (listed below) are maintained

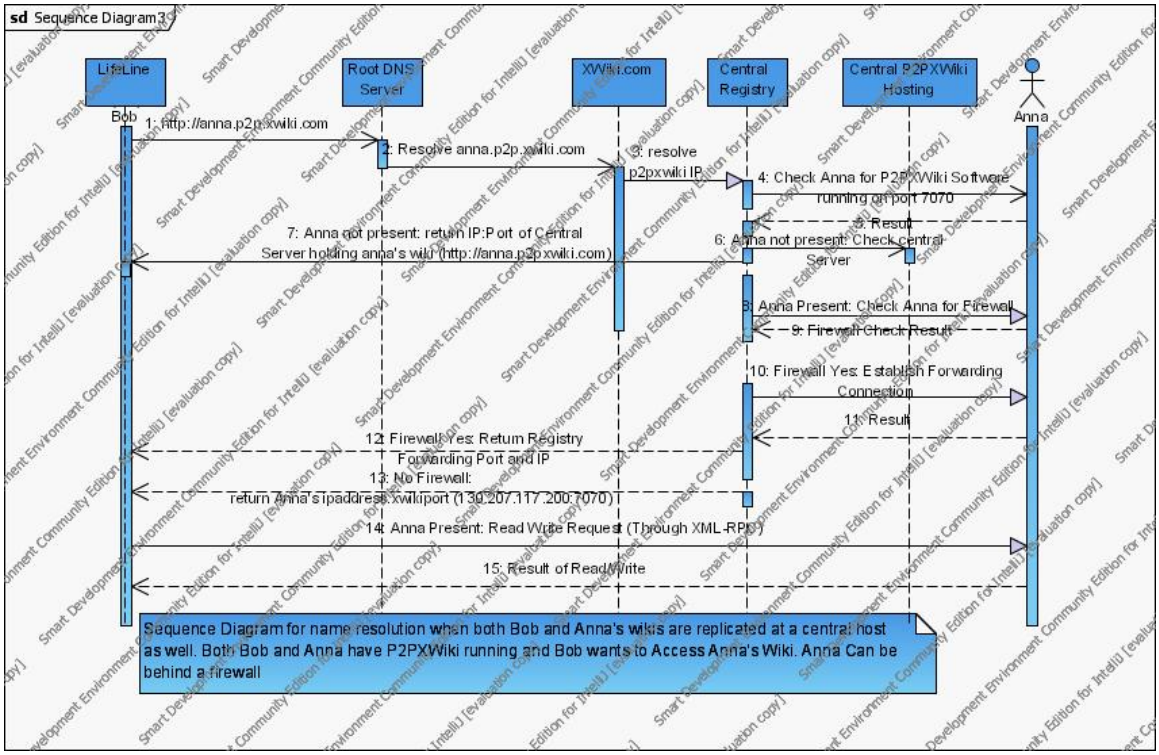
        - none of the changes to a wiki are lost.
        - the master and the central hosting service agree on the content of the wiki when there is no writer.
- **Release 0.3: Disseminating a wiki among peers**
- The wiki dissemination and discovery in a p2p network providing readonly copies
  - Handle Firewall issues where peers may be behind firewalls.
  - Extensions to XML-RPC protocol
    - make sure all save apis are available so we can pass through firewalls through the forwarding connection
    - A full replication API (get a list of documents since date XXX, multiple updates to documents, etc...)
  - The master controls which replicas are going to be consistent and redirects read requests to one of the peers.
  - All write requests are still handled by the master.
- **Release 0.4: Allow direct sharing of wiki among peers (read only) even when master is not present**
- peer can directly access a wiki from any peer who has a fully replicated copy for read only.
  - each peer is responsible for pulling updates from the master for the wikis that are replicated.
  - All the writes to a wiki still goes to the master.
- **Release 0.5: Allow peers to edit the wiki in the absence of Master**
- All the edits are saved for later synchronization when the master comes offline.
  - The edits still remain local. No other peer, except the writer, sees these.
- **Release 0.6: Allow TRANSFER of Master rights to some specific nodes (which won't go offline) and also allow revocation**
- consistency among group of masters
  - this allows more availability. The group of nodes can be part of the hosting servicew.
- **Release 0.7: Allow sharing of Edits to the wiki by transfering master rights to some other peers.**
- This will allow a group of people not connected to the internet to share content and edit them.
  - Consistency handling when the group comes online.
- **Release 0.8: Replicating the Registry**
- allow other peers to also volunteer to store the registry.
  - consistency and integrity of multiple peer registries.
- **Release 0.9: P2P Presence: With user interface integration**
- **Release 1: Security - handling wiki hijacking**
- allow correct wiki to be retrieved in a decentralized manner even when peers are untrusted - perhaps using some public/private key
  - allow selective content sharing (only the ones you want are replicated)
- **SOC Target: Somewhere between 0.2/0.3. Perhaps 0.1 will be the deliverable with progress showing towards 0.2/0.3.**

Use Case Diagrams

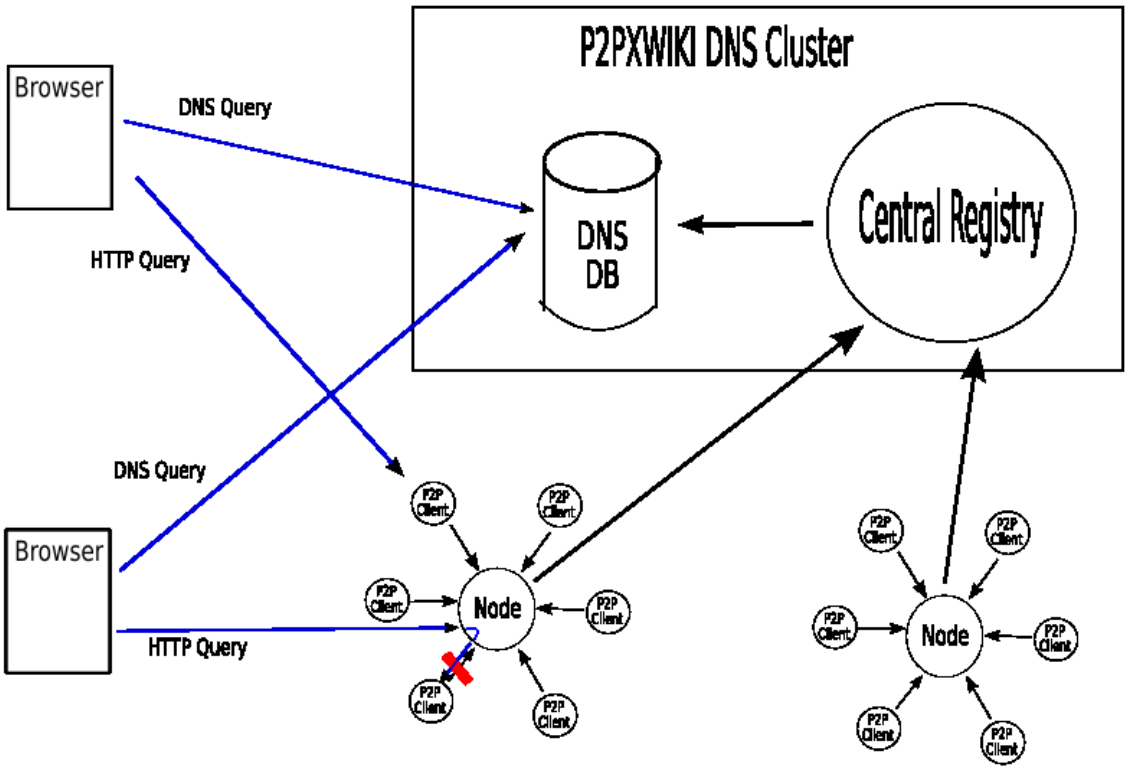
Some UGLY diagrams with use cases for name resolution demonstrating lower level technical architecture for version 0.1 of p2pxwiki. Better software is needed to develop these diagrams. These are probably not necessary, and also time consuming to draw, so I will avoid it and use simple diagrams !


sd Sequence Diagram1
Bob
Root DNS Server
XWiki.com
Central Registry
Anna
1: http://anna.p2p.xwiki.com
2: Resolve anna.p2p.xwiki.com
3: resolve p2pxwiki IP
4: Check Anna for Firewall
5: Firewall Check Result
6: Firewall Yes: Establish Forwarding Connection
7: Result
8: Firewall Yes: Return Registry Forwarding Port and IP
9: No Firewall: return Anna's ipaddress:xwikiport (130.207.147.200:7070)
10: Read Write Request (Through XML-RPC)
11: Result of Read/Write
Sequence Diagram when both Bob and Anna have P2PXWiki running and Bob wants to Access Anna's Wiki. Anna Can be behind a firewall


sd Sequence Diagram2
Bob
Root DNS Server
XWiki.com
Central Registry
1: http://bob.p2p.xwiki.com
2: Resolve bob.p2p.xwiki.com
3: resolve p2pxwiki IP
4: Return bob's local IP address (127.0.0.1:7070)
5: Read Write Request
Sequence Diagram for name resolution when Bob Wants to access a local wiki



Proposition for the P2P Name Resolution



Dev.Design Issues (en)
Creator: XWiki.bikash Date: 2005/07/19 09:33
Last Author: xwiki:XWiki.LudovicDubost Date: 2005/08/02 10:17