

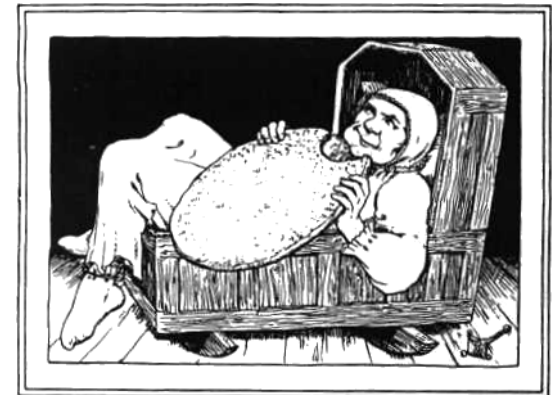


For your consideration...

SMB in a Chewable Size

Defining a Low Level API

Christopher R. Hertel
wekalo & Samba Team
September 2018
SNIA SDC





Quick Introductions





Introductory esquenessism

Me:

- Samba Team Elder
- SMB Wizard with...



WEKA.io

The opinions expressed are my own and not necessarily those of my employer, my colleagues, my spouse, my spirit familiar, the Internet Voices, or the monster in the closet.

A scenic landscape featuring a vibrant green field under a bright blue sky filled with large, fluffy white clouds. In the foreground, there are tall, dry, brown stalks of grass or reeds. Scattered across the green field are several round hay bales. In the distance, a single tree stands on a slight rise, and a few more trees are visible on the horizon.

SMB Background

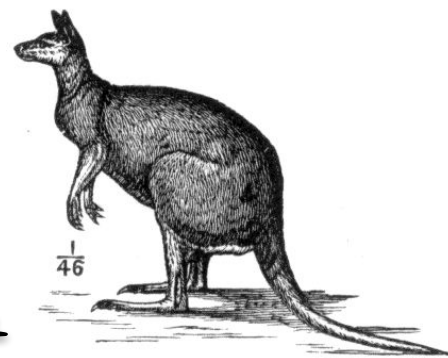


Taxonomy



Alpaca, p. 63.

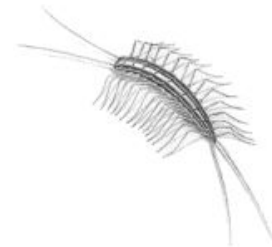
Block
File
Object



Kangaroo, p. 1177.

Three general categories for network data storage.

- Obviously **$\${userPref}$** is *the best*
- Two predominant Network **File** Protocols:
 - NFS (Network File System)
 - SMB (Server Message Block)





SMB is a File Protocol

SMB presents:

- Shares
 - Directories
 - Files



A network extension for DOS, OS/2, and Windows File Systems for over 3 decades.

- Also carries system calls,
- Named Pipe Operations (IPC), and
- Remote Procedure Calls (RPC).



SMB is a Network Transport

SMB is Infrastructure

- A Reliable, Scalable, Authenticated, Signed, Sealed, Network Connection
 - We've already mentioned RPC
 - There's also RSVD
 - Block I/O for Storage Spaces Direct

Think of SMB as a Transport.





SMB History

What are SMB1, SMB2, and SMB3?

SMB1: The Original

- ❖ Created in the early 1980's by IBM
- ❖ Further development by 3Com, IBM, Intel, and Microsoft
 - For PC-DOS, MS-DOS, and OS/2
 - Dialect updates released with Operating System Updates
- ❖ Ported to Windows NT
 - Final Dialect! "NT LM 0.12"





SMB History

What are SMB1, SMB2, and SMB3?

SMB2: Introduced with Vista!

- ❖ An Entirely New Protocol
 - 1/4th as many message types
 - No DOS or OS/2 Baggage
- ❖ Similar Design
 - Familiar to SMB1 Developers
- ❖ Silently Negotiated
 - Users neither knew nor cared





SMB History

What are SMB1, SMB2, and SMB3?

SMB3: Marketing Upgrade!

- ❖ An SMB2 Dialect; not a new Protocol
 - Originally planned as 2.2
 - Message structures are the same
- ❖ Adds Speed, Scalability, Reliability
 - RDMA
 - Multichannel
 - Persistent Handles

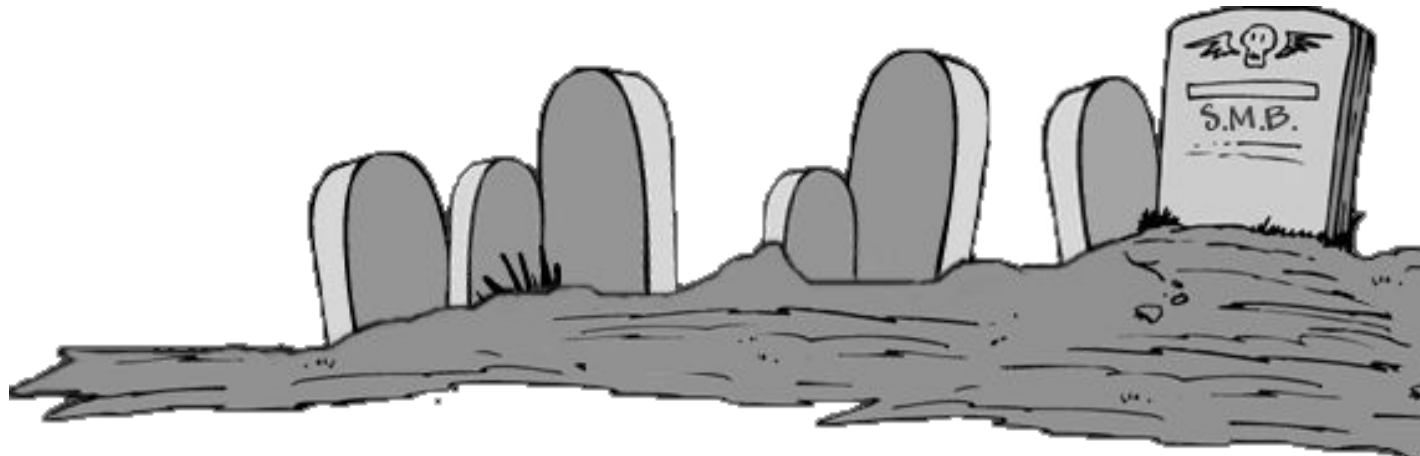




SMB History

What are SMB1, SMB2, and SMB3?

SMB1 is dead.

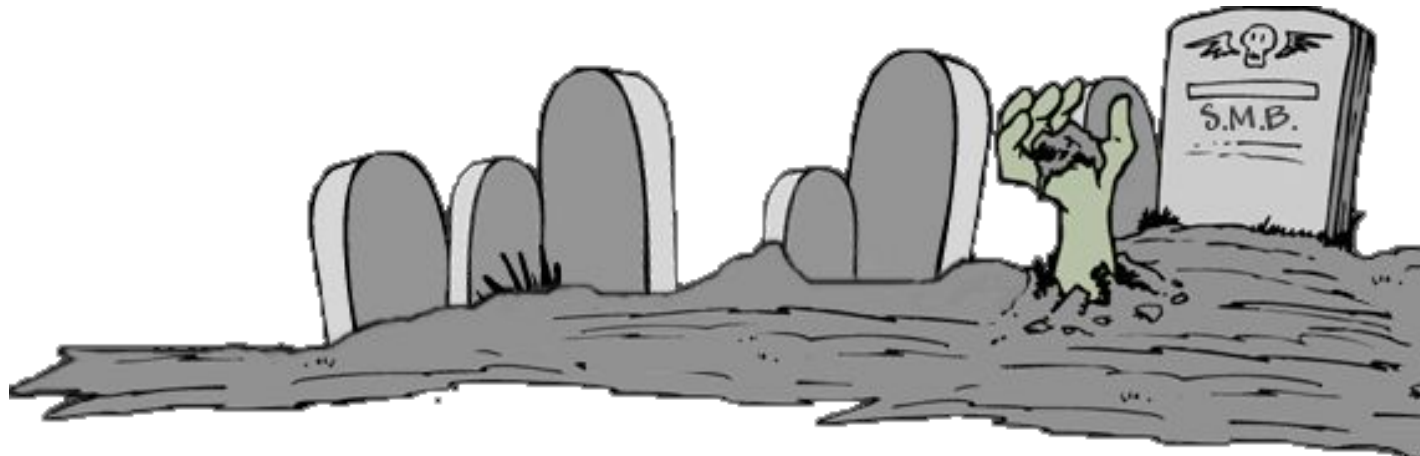




SMB History

What are SMB1, SMB2, and SMB3?

CIFS is Deader.





SMB Stability

How standard is SMB?

- It's **Not!**
- SMB1 was Undocumented for Years.

However...

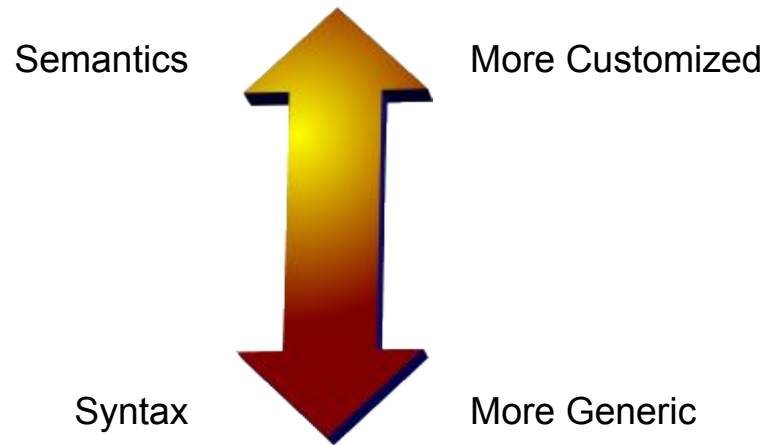
- Backward Compatibility is a Business Requirement
 - It's Now Documented
 - It's Entwined with the Infrastructure
- 



*Start Paying
Attention Now*

An SMB2/3 Protocol Engine

Consider the SMB "Protocol Stack"



- ❑ SMB2/3 message handling can be broken down into layers
- ❑ Layers can be interconnected by Programming Interfaces
- ❑ You know, like the OSI stack



An SMB2/3 Protocol Engine

By leveraging the SMB2/3 message handling hierarchy, we can...

- Isolate message parsing/packing
- Handle message syntax errors
- Identify and manage base-level state





An SMB2/3 Protocol Engine

What I am proposing here is a:

- ❑ Standard,
- ❑ Low level,
- ❑ SMB2/3 Messaging API





An SMB2/3 Protocol Engine

Why?



- ★ Implement it as a library
- ★ ...or perhaps as a device
- ★ ...or both
- ★ Implement it in offload cards:
 - Handle the Underlying Transport (TCP, RDMA)
 - Multichannel
 - Signing and Sealing

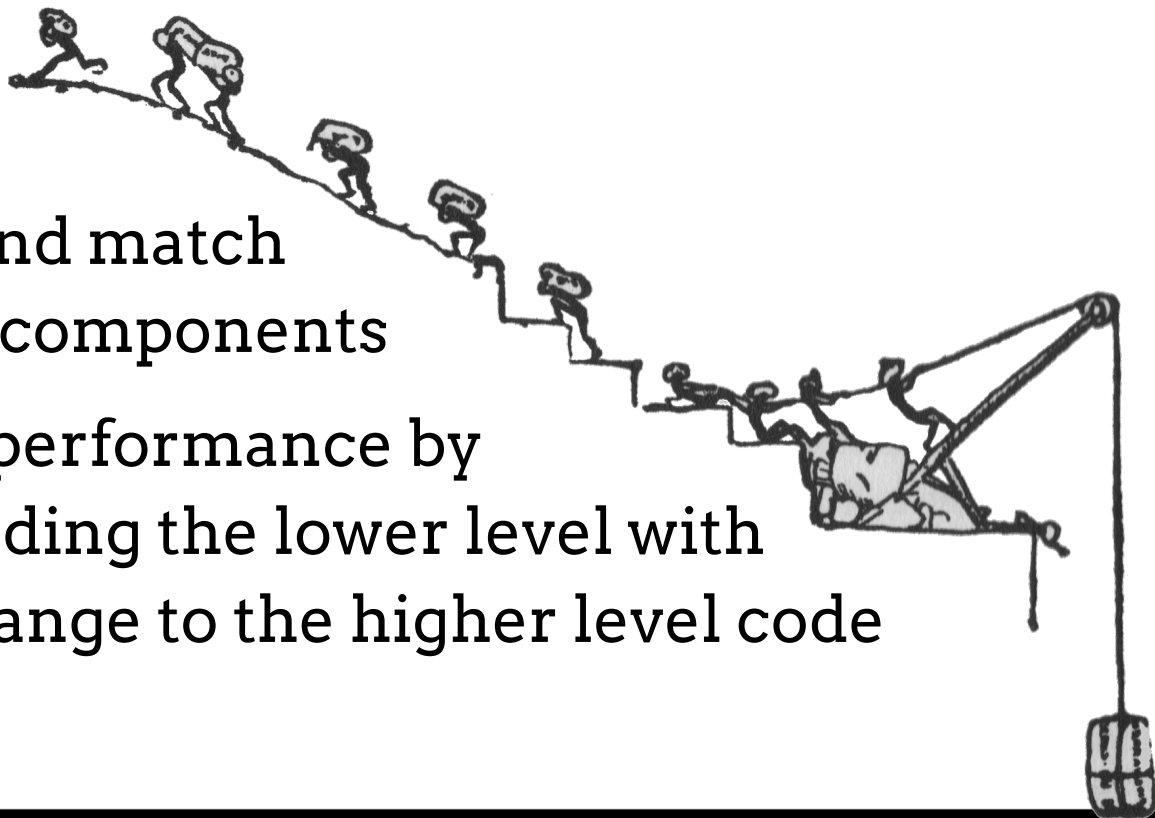


An SMB2/3 Protocol Engine

If done right...

- ⊙ Separates the Semantic from the Syntactic layers

- ⊙ Mix and match stack components
- ⊙ Gain performance by upgrading the lower level with no change to the higher level code





An SMB2/3 Protocol Engine

SMB2 Message Handling Hierarchy

2. **Syntax** (SECTION 2)
Packet Parsing and Packing
3. **Semantics** (SECTION 3)
What does it all mean?
3. **State** (SECTION 3)
**Maintain and Transition
per Input**





An SMB2/3 Protocol Engine

NOTHING IS CARVED IN STONE



...BUT HERE'S WHAT I HAVE IN MIND.



An SMB2/3 Protocol Engine

SMBopen()

- Open a handle to an SMB2/3 message "device".

SMBcreate()

- Create a handle by binding a socket to an SMB2/3 message library.
- Allow stacking, to support new dialects.

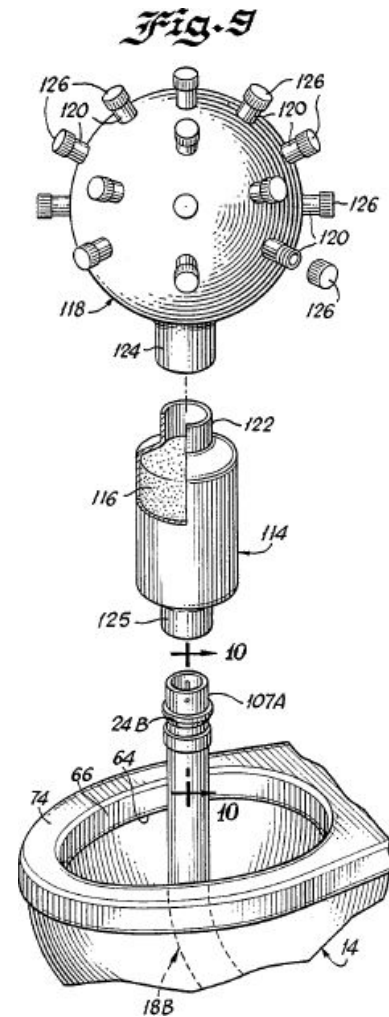
SMBclose()

- Close an smb2/3 message handle, freeing resources.

An SMB2/3 Protocol Engine

The design can be informed by the socket model.

We'll need equivalents to connect (2) and accept (2), etc.





An SMB2/3 Protocol Engine

SMBsend()

- Given a data structure, compose and send a correct SMB2/3 message.
- Validate the message before sending.

SMBrecv()

- Upon receiving a message, parse it.
- Handle obvious protocol errors.
- Provide the parsed message to the caller.

We may also need `select(2)` / `poll(2)` equivalents to know when messages are ready.



An SMB2/3 Protocol Engine

SMBsetopt()

- Set engine-internal parameters
 - ...such as which dialects we support
 - ...or which capability bits
 - ...or enable/disable crypto options

SMBgetopt()

- Retrieve engine-internal parameters
 - Ask which dialects the engine supports
 - Retrieve engine statistics

Obviously parallel to `get/setsockopt(2)`.



The Plan

This should be a win for:

- SMB Implementers
 - Open Source and Commercial
- Hardware Vendors
 - Cards
 - Platforms
- Device Driver Developers

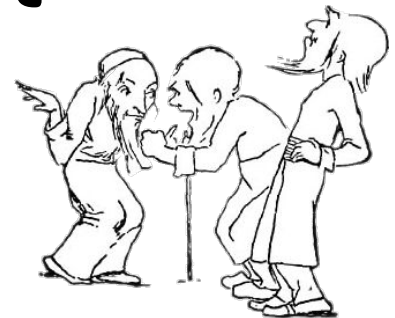




The Plan

To Create a Standard, we need:

- **A rough-draft spec from which to start**
- **A Committee**
- **A reference implementation**
- **An organization to provide structure and support**



The End

