WSON Routing WG Drafts

1. Routing and Wavelength Assignment Information Model for WSON
2. General Network Element Constraint Encoding for GMPLS Controlled
3. Routing and Wavelength Assignment Information Encoding for WSON
4. OSPF Enhancement for Signal and Network Element Compatibility for WSON

Authors/Contributors

- Greg Bernstein (Grotto Networking)
- Diego Caviglia (Ericsson)
- Ander Gavler (Acreo AB)
- Young Lee (Huawei)
- Dan Li (Huawei)
- Wataru Imajuku (NTT)
- Jonas Martensson (Acreo AB)
- Itaru Nishioka (NEC Corp.)
- Jianrui (Rebecca) Han (Huawei)
- *Plus a whole lot of folks on the CCAMP list and at the last four years of meetings.*

WSON Draft Development Process (I)

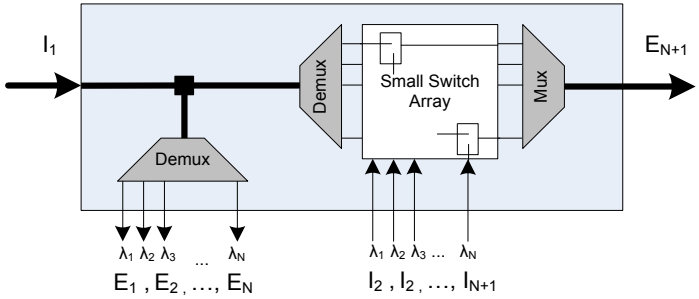
- First CCAMP presentation July 2007 (Chicago)
- Status and copies of all supporting materials were made easily available from web site
- Vendors, Carriers, Researchers, experienced GMPLS developers all contributed to drafts
- Evaluated models against many test cases
- Complied with CCAMP WG direction concerning structuring of drafts

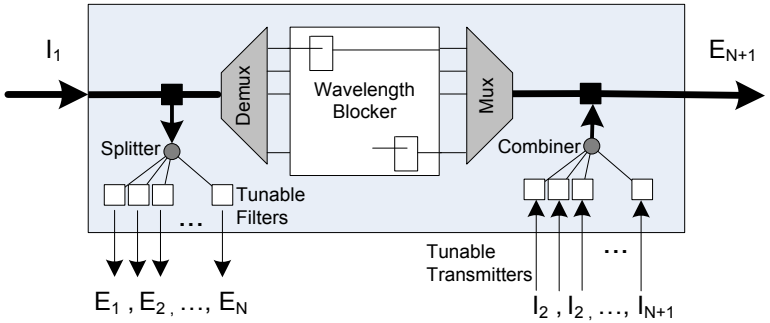
WSON Draft Development Process (II)

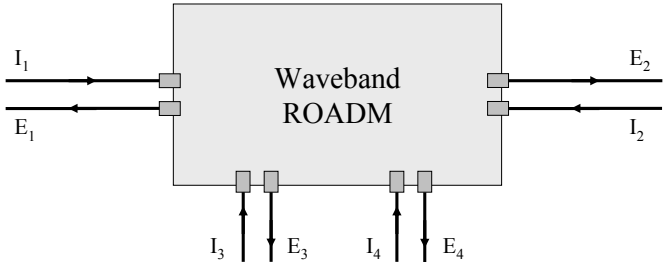
- Many supplemental tutorials and presentations at IETF meetings, Optical Fiber Conference, IPoP, and OSA/IEEE Journal articles, e.g.,
  - WSON Editing & Discussion meeting, 73rd IETF, MN, Nov., 2008 (models with resources introduced and inputs solicited).
  - G. Bernstein and Young Lee, "Extending GMPLS/PCE for use in Wavelength Switched Optical Networks," in *Conference on OFC/NFOEC, 2008.*
  - G. Bernstein and Young Lee, "Overview of the Path Computation Element (PCE) and its application to Wavelength Routing," presentation at Optical Routing Workshop OFC, 2008.
  - G. M. Bernstein, Y. Lee, A. Gavler, and J. Martensson, "Modeling WDM Wavelength Switching Systems for Use in GMPLS and Automated Path Computation," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 1, no. 1, pp. 187-195, Jun. 2009.

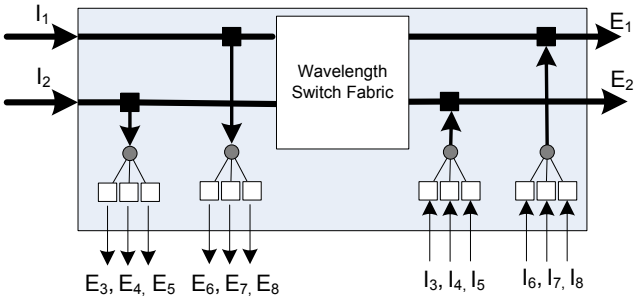
WSON Draft Development Process (III)

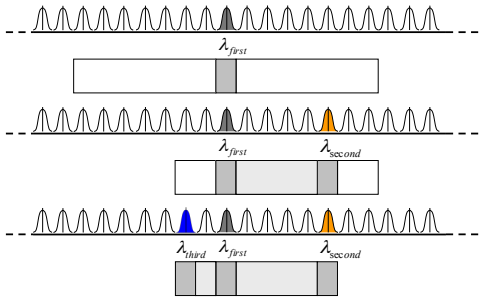
- Many Test Cases (w/o resources) e.g.:

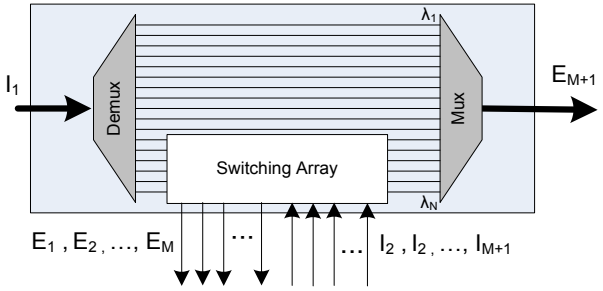








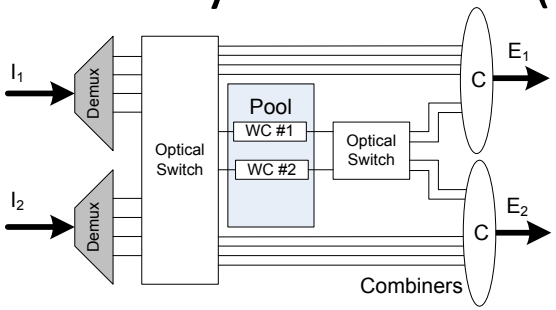


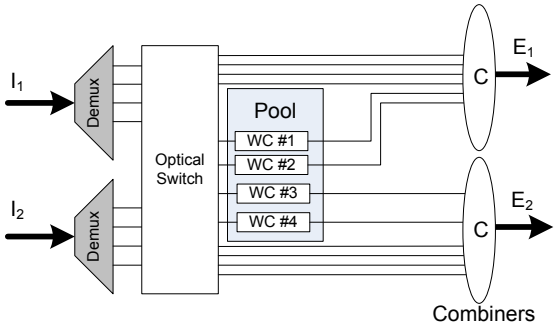


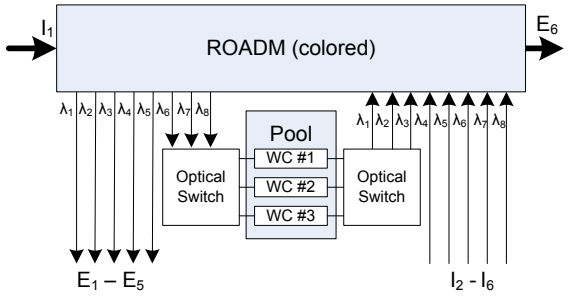
Etc…

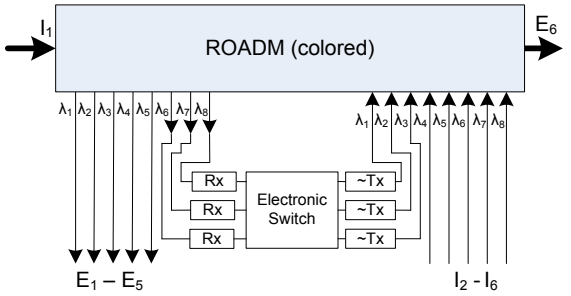
WSON Draft Development Process (IV)

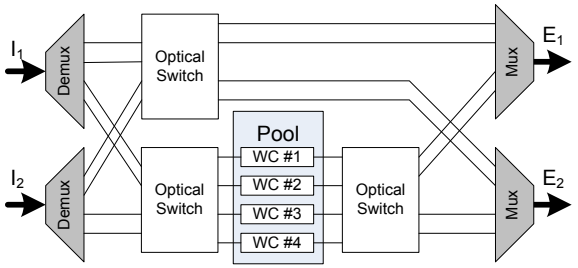
- Many Test Cases (with resources) e.g.:

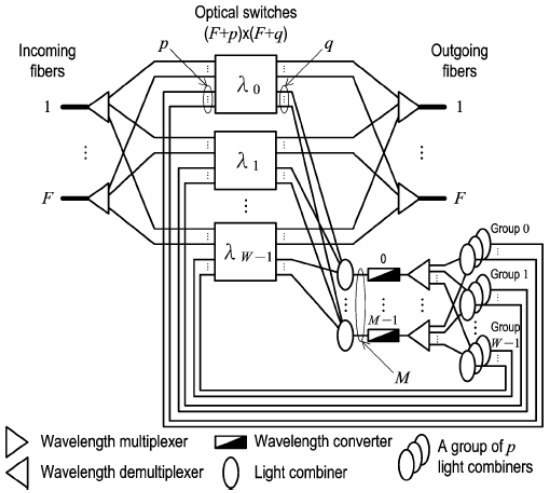












Etc…

WSON Draft Development Summary

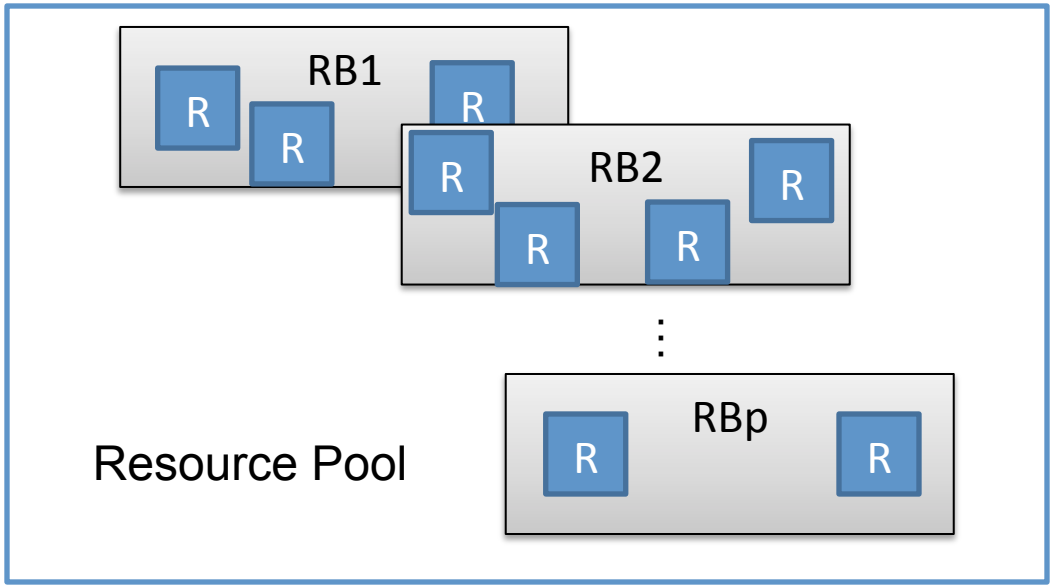
- Many person years of development
- Open process with wide review of approach, beyond CCAMP and IETF
- Call for inputs and evaluation of models with resources went out in November 2008
- Any changes to documents at this point should be based on a well ***demonstrated problem***, with clearly furnished minimalistic example
- Proposed ***changes should be minimalistic*** to avoid breaking existing mechanisms.

Changes Requested on List

1. "Introduction of the Resource Pool entity inside the model, which allows the definition of several resource entites per node independantly floodable."
   - Current encoding has this capability.
2. "Use of the connectivity matrix defined inside the node entity in order to describe connectivity constraints between node-external links and the resource pools."
   - No size or stability benefit from change, This was split out at 74th IETF meeting March 2009.
3. "Reduction of the scope of Resource Block Information, to keep only resource/device description (moved the number of devices away)."
   - This is an optional ***static*** sub-sub-TLV, hence you don't need to use it. Dynamic information is in the *resource pool state sub-TLV*

Resources, Blocks, ***Sets*** and Pools

R = resource
RB = resource block



Resource sets:
(RB1, RB2),
(RB2, RB3, RB4),
Etc…

- Simple hierarchy:
  - The ***resource pool*** is partitioned into ***resource blocks*** containing individual ***resources***.
- Efficient and general encoding:
  - Resources blocks are combined into ***sets*** for encoding of common properties.

Change Request 3 (I)

- “Reduction of the scope of Resource Block Information, to keep only resource/device description (moved the number of devices away).”
- “MOTIVATION:
  - a/ to share resource description for all the (same devices) blocks (of a node), then decreasing the total size of information.
  - b/ to create an independent flooding entity holding all the resource descriptions (which are static), the decreasing the size of updated information.”

Change Request 3 (II)

- What is this item?
  - Section 4 of WSON encoding is concerned with properties of resources; Section 4.1 introduces the ***Resource Block Information sub-TLV***
  - Section 4.6 introduces the ***Processing Capabilities List sub-sub-TLV*** which includes the *option* of specifying the ***Number of Resources within the block***
- What is it for?
  - In many cases multiple resources come bundled together, e.g., on a line card. This tells us the number of resources in this "block".

Change Request 3 (III)

- Is it a dynamic or static quantity?
  - This is a static quantity.
- How does this differ from the ***Resource Pool State sub-TLV***?
  - This is a dynamic quantity that tells you the number of resources available in a particular RB pool set. It does not tell you the number in use/out of service.
- Do I have to use it?
  - No, it is an optional field
- Can we remove it from the draft?
  - No. A number of optimization algorithms want to know the total number of resources available at each node.

Change Request 2 (I)

- “Use of the connectivity matrix defined inside the node entity in order to describe connectivity constraints between node-external links and the resource pools.”
- “MOTIVATION:
  - a/ to gather static information inside node entity (for OSPF-TE inside a LSA never flooded upon LSP updates).
  - b/ to limit the number of connectivity representations introduced by current extensions (draft-ietf-rwa-info proposes similar TLVs in different LSAs)”

Change Request 2 (II)

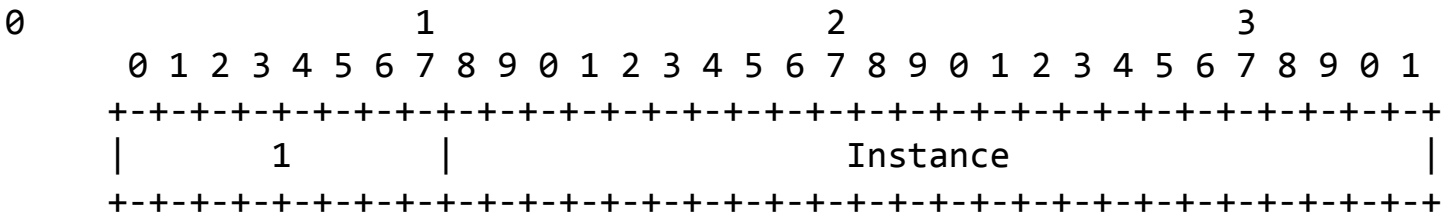
- What is this about?
  - The ***WSON info model draft*** discusses connectivity between externally visible ingress and egress ports (the ***connectivity matrix***) and connectivity between externally visible ingress and egress ports and internal resources (***resource accessibility***). These use similar mechanisms.
- If these use similar mechanisms why aren't they combined?
  - We initially had these combined but were asked to separate out concepts and mechanisms with general applicability from those specific to WSON. This decision was discussed at a number of IETF meetings and finalized at the 74th IETF meeting in San Francisco, March 2009. This is why there are two encoding drafts (general constraints and WSON specific).

Change Request 2 (III)

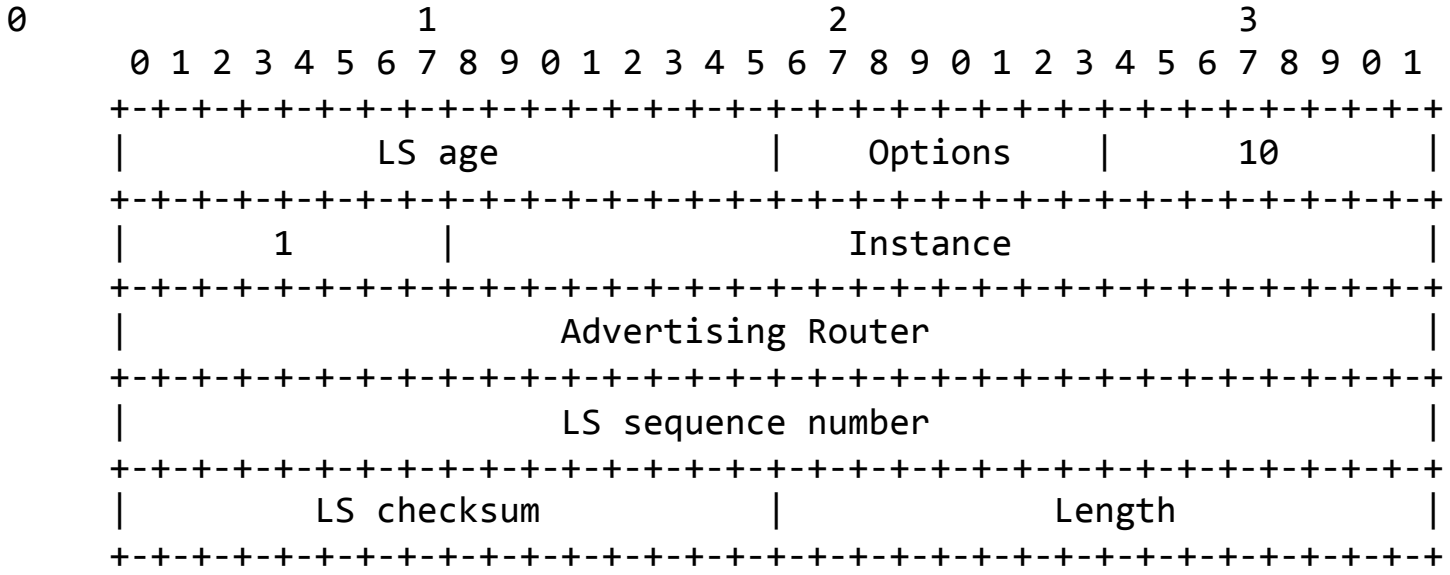
- Does this result in any significant space savings or expansion?
  - No. The same basic information needs to be conveyed whether it is split into two different top level TLVs or in just one.
- What is the implications for static versus dynamic information?
  - This is an orthogonal issue from this particular encoding choice. We recommend that static and dynamic information should be flooded separately. OSPF-TE provides a good mechanism for this. See the following slides for a review of the basic mechanisms.

Updating WSON info with OSPF-TE (I)

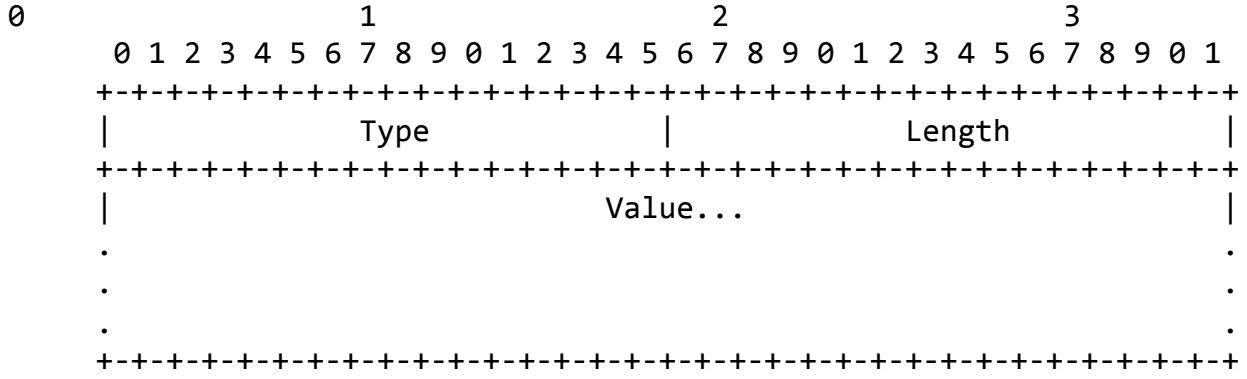
- OPSF Opaque LSA and Optical Node TLV
  - All of GMPLS uses the OSPF-TE (RFC3630) ***Traffic Engineering LSA***.
  - "The LSA ID of an Opaque LSA is defined as having eight bits of type data and 24 bits of type-specific data. The Traffic Engineering LSA uses type 1. The remaining 24 bits are the Instance field, as follows:"
  - "The Instance field is an arbitrary value used to maintain multiple Traffic Engineering LSAs. A maximum of 16,777,216 Traffic Engineering LSAs may be sourced by a single system.



OSPF-TE LSA (RFC3630)



"The LSA payload consists of one or more nested Type/Length/Value (TLV) triplets for extensibility."



Updating WSON info with OSPF-TE (II)

- LSAs are the smallest flooding/update unit
  - You pick what you want to put within a particular LSA instance within the constraints of the TLV hierarchy and rules, e.g., one top level TLV per LSA (RFC3630)
- Currently Defined and Proposed Top Level TLVs
  - RFC3630: Router Address, Link
  - RFC5329: Router IPv6 Address, RFC4203: Link Local
  - RFC5786: Node Attribute
  - WSON OSPF extensions: Optical Node Property TLV

Updating WSON info with OSPF-TE (III)

- All of the WSON specific info goes into an Optical Node TLV?
  - Yes.
- How do I keep this TLV from getting too large? How do I separate dynamic and static information?
  - Use multiple TE-LSA instances each with a Optical Node TLV containing one or more sub-TLVs.
  - This is why we never mix static and dynamic information in the same sub-TLV
  - This is also why we break information up into sub-TLVs

Change Request 1 (I)

- “Introduction of the Resource Pool entity inside the model, which allows the definition of several resource entites per node independantly floodable.”
- “MOTIVATION: to decrease the size of flooding upon LSP changes (setup or tear down). (Resource Pool = group of resource blocks with same connectivity constraints).”

Change Request 1 (II)

- Current model has the containment hierarchy of:
  - Resource, Resource Block, Resource Block Set, and Resource pool.
- The ***Resource Block Set*** concept serves this function.
  - ***A general group of resource blocks with similar properties of any kind***
  - No need for new terminology.

Resource Block Set Usage

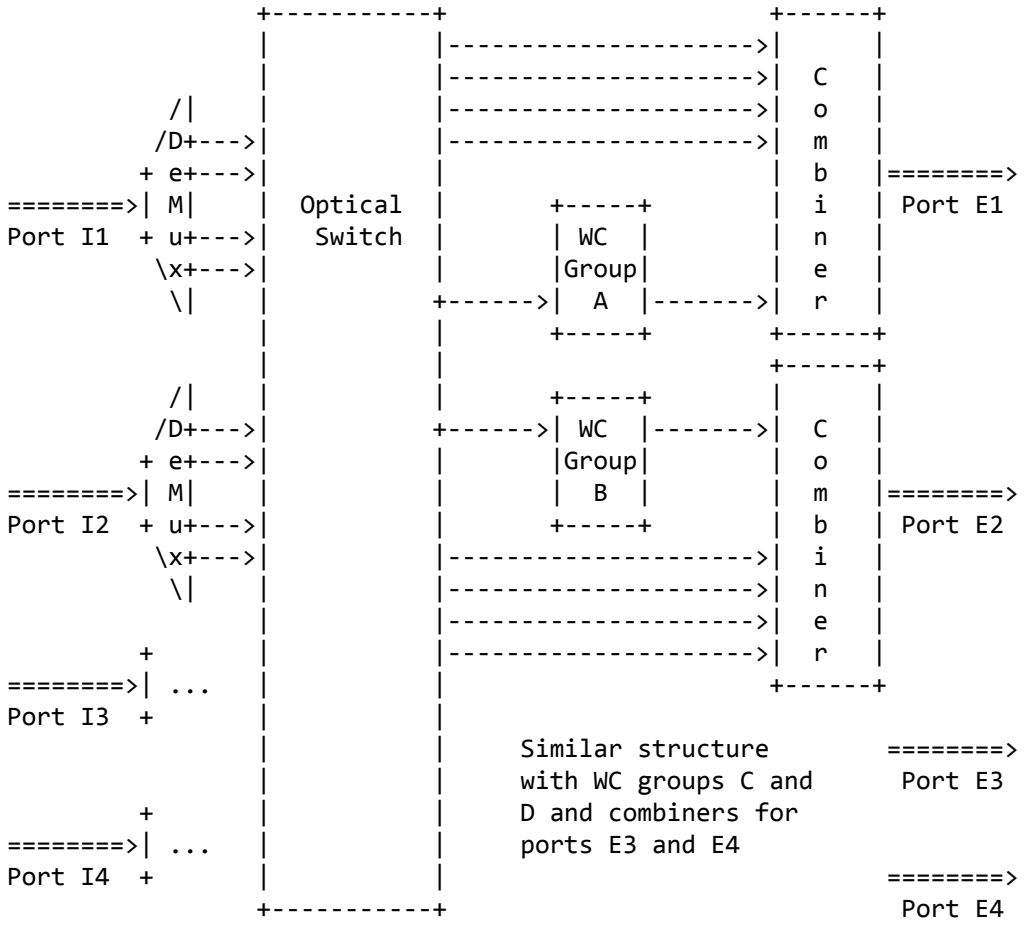
These sub-TLVs can be placed in one or more Optical node property TLVs (in one or more Opaque LSAs), all independently floodable.

- Resource pool accessibility sub-TLV
- Resource block wavelength constraints sub-TLV
- Resource block information sub-TLV
  - Fixed this in March 2011 to use resource set
- Block shared access wavelength availability sub-TLV
- Resource pool state sub-TLV

Static

Dynamic

Example: Many types of regens (I)



Numbers: 4 ports, 200 wavelength converters, 10 types of converters, random assignment of converters to groups.

Example: Many types of regens (II)

Accessibility sub-TLV:

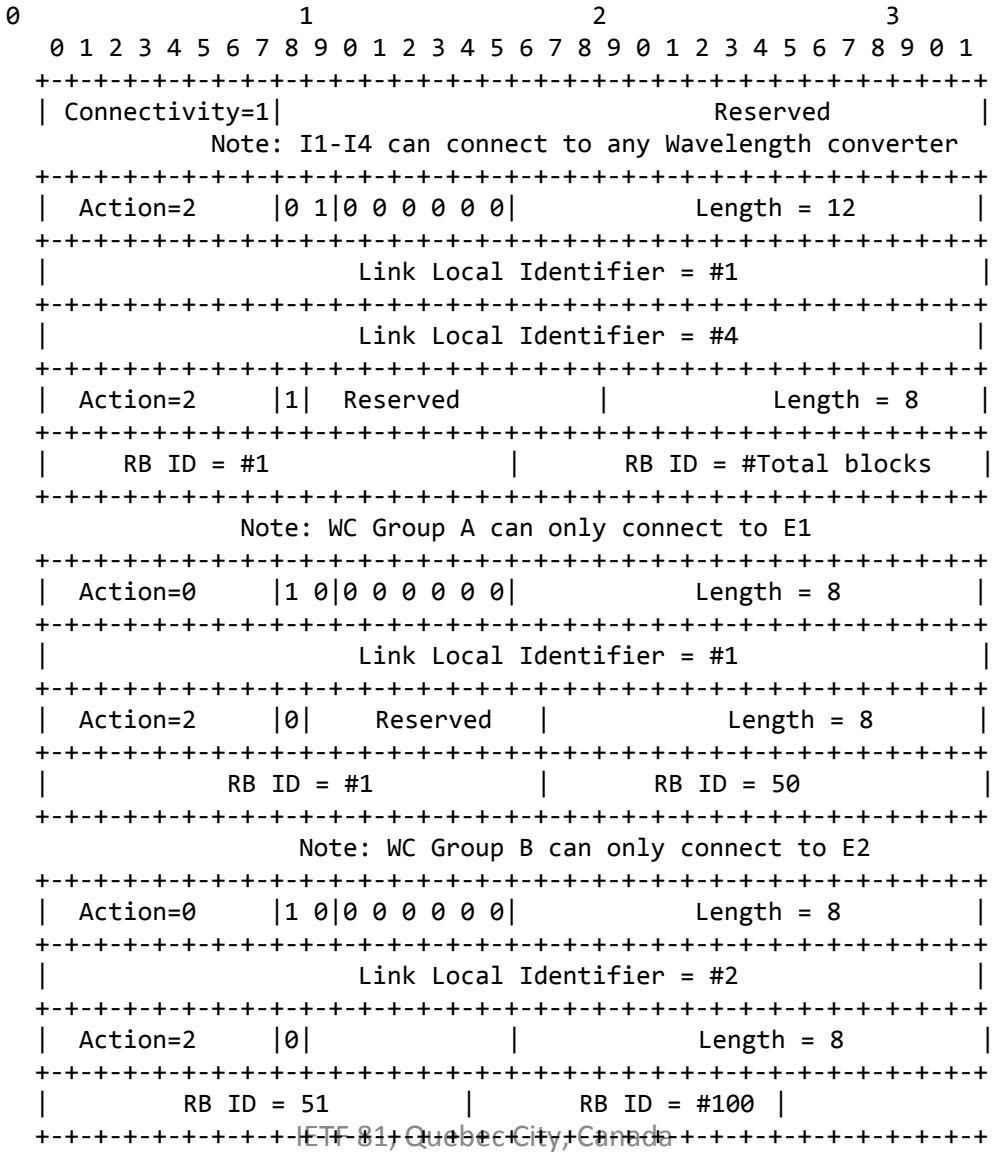
(a) Linear numbering

(b) 50 converters per group

(c) Use ranges to describe group

(d) Total words: 6+4x4 = 22 → ***88 bytes***

(e) All information is static

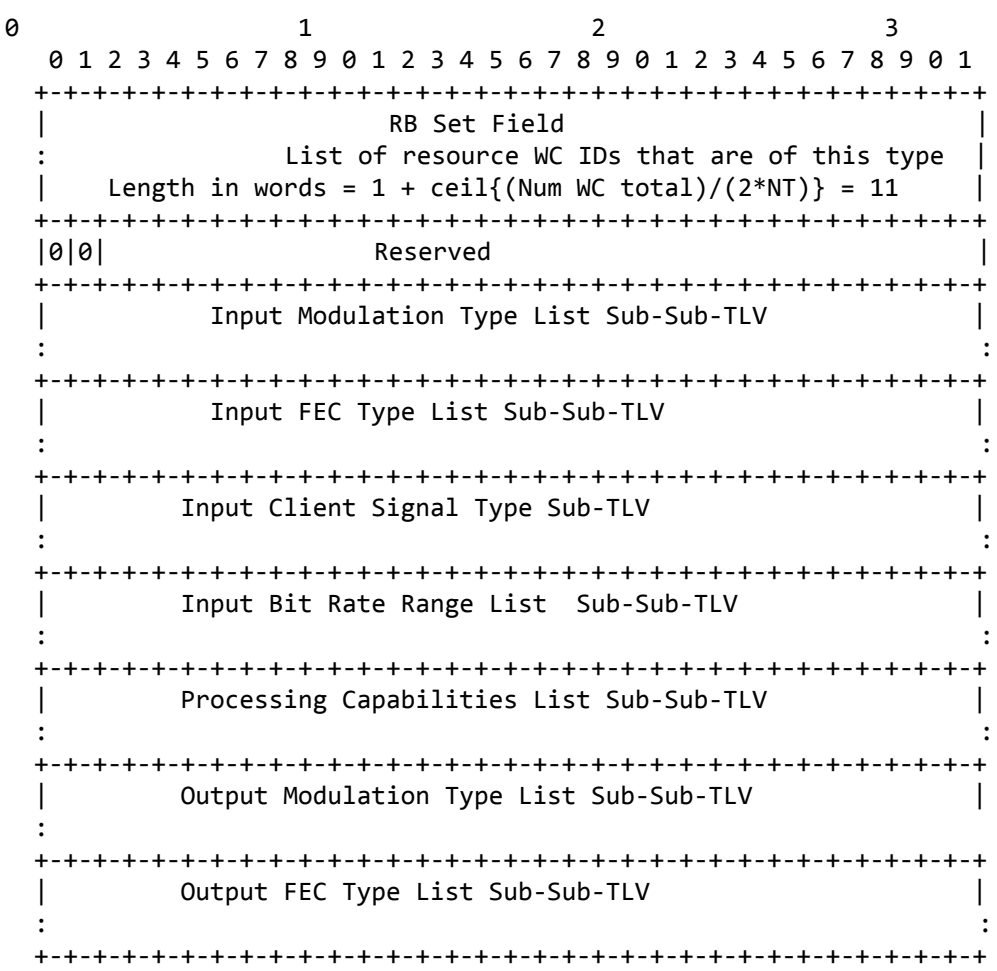


Not shown WC Groups C and D

Example: Many types of regens (III)

Resource Info sub-TLV:

(a) Random numbering
(b) 20 converters per type
(c) Use list to describe groups
(d) Assume 32 words for info fields + 11 words for list = 43 words per type = 172 bytes per sub-TLV
(e) Need 10 of these sub-TLVs
(f) Sub-TLVs can be sent separately
(g) All information is static



Example: Many types of regens (IV)

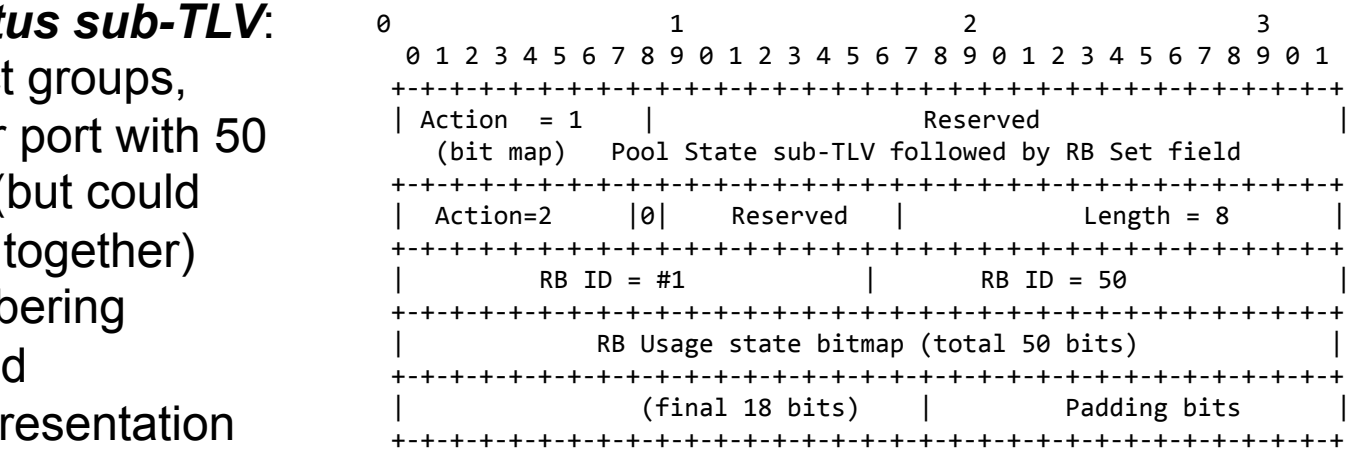
Resource Status sub-TLV:

(a) Four distinct groups, i.e., one per port with 50 resources, (but could grouped all together)

(b) Linear numbering already used

(c) Bit map representation for resource status

(d) Total of 5 words (20 bytes) for each sub-TLV



Similarly for resources 51-100, 101-150, 151-200

Example: Many types of regens (V)

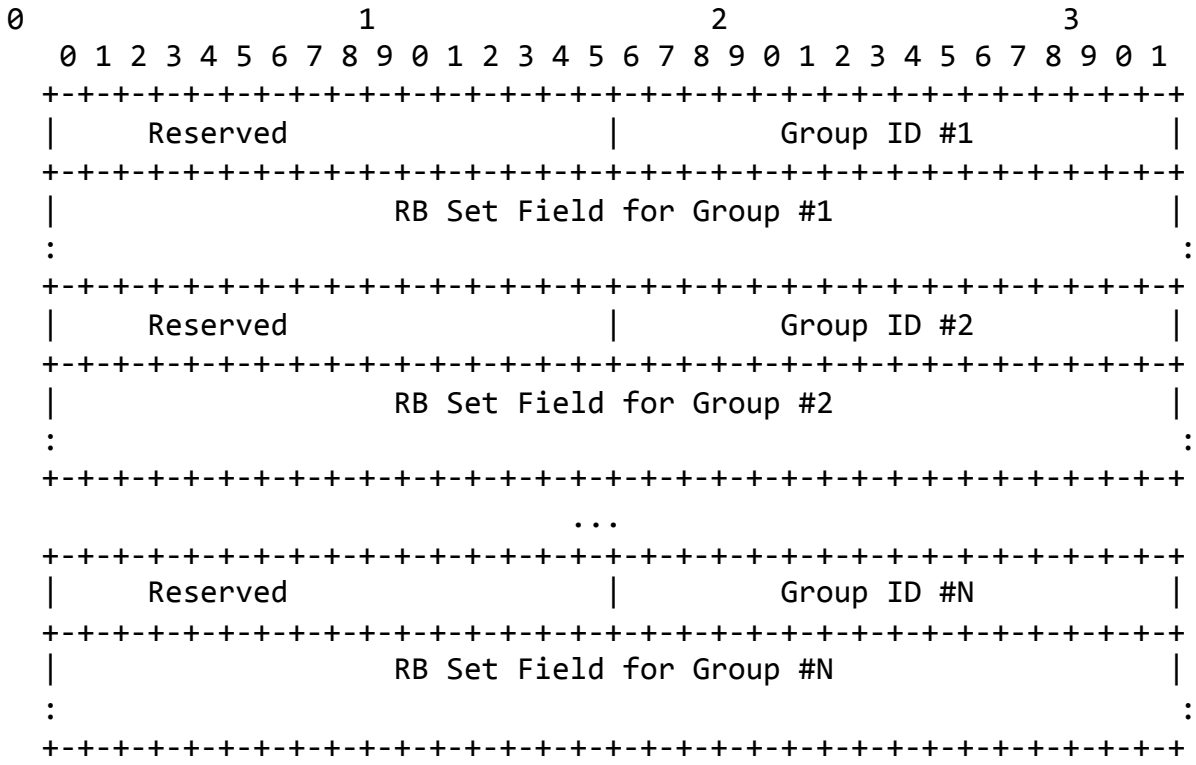
- All information in separate sub-TLVs can be sent in separate Traffic Engineering LSAs with different instance numbers
  - Accessibility sub-TLV (88 bytes) –static--
  - Resource Info sub-TLVs (10 @ 172 bytes each) –static --
  - Resource Pool state sub-TLVs (4 @ 20 bytes each) – dynamic –
- LSA, TLV, and sub-TLV overhead not included in estimates.

Concern expressed on list: numbering of resource blocks

- Potential Issue:
  - I've got very complicated optical nodes and I can't guarantee the linear assignment of RB identifiers, so can't always use ranges for dynamic updates…
- Potential Solution
  - Add ***resource group definition sub-TLV***
  - Add ***predefined group identifier*** option to RB Set field.

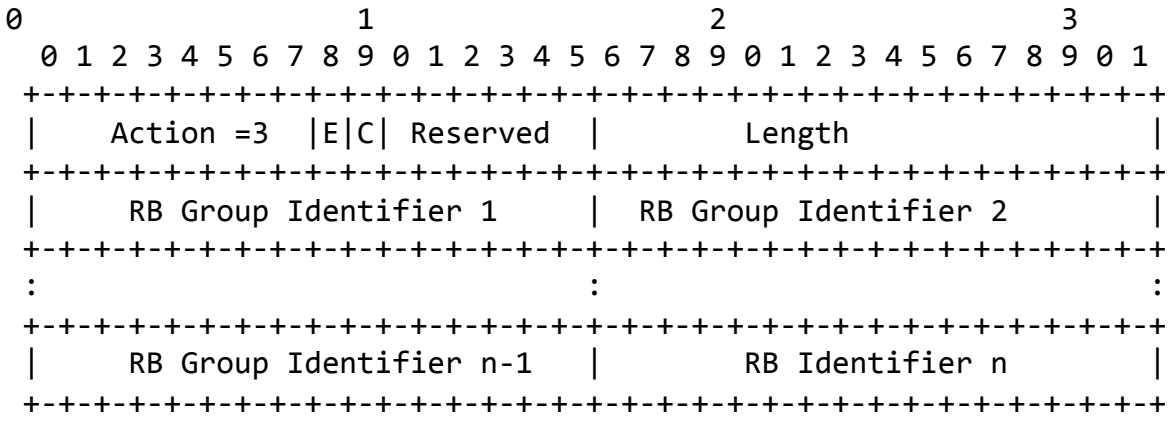
Numbering of resource blocks (II)

Potential new *Resource Group Definition Sub-TLV*



Numbering of resource blocks (III)

Potential new *RB Set Field Action*



Action = 3 – Predefined Group Identifier
Indicates that the field contains one or more resource block group identifiers. This is potentially useful in cases where a very long resource block sets may be reused, such as in resource state update messages.

But is this really needed?

This is an extra complication and no realistic example has been given that justifies this addition.

Other Concerns Expressed On List

- Stability of resource block identifiers across life time of box.
  - Three aspects here:

  a) your systems HA architecture (not standard),

  b) routing information and its configuration, (not standard)

  c) signaling state recovery (standardized)