

11100 Johns Hopkins Road Laurel, MD 20723-6099

> Edward J. Birrane, Ph.D. Johns Hopkins University, Applied Physics Laboratory (JHU/APL) Edward.Birrane@jhuapl.edu (W) 443-778-7423

#### **DTN Network Management**

# History

- Examined uniqueness of the problem, 2011-2013
  - Some early pubs defining the problem as related to DTN
    - Birrane, E, & Cole, R. (2011). Management of Disruption-Tolerant Networks: A Systems Engineering Approach.
    - E. Birrane, S. Burleigh, V. Cerf, "Defining Tolerance: Impacts of Delay and Disruption when Managing Challenged Networks,"
    - E. Birrane, H. Kruse, "Delay-Tolerant Network Management: The Definition and Exchange of Infrastructure Information in High Delay Environments"

#### - Reviewed popular engineering approaches

- Autonomous fault protection schemes
- Mobile code and scripting schemes
- Spacecraft telemetry schedules
- Deterministic rule-based expert systems
- Delay-Tolerant Network Management Protocol (DTNMP) 2013
  - Published to DTNRG, Initial implementation by NASA
  - Utility outside of NASA network management
- Renamed as Asynchronous Management Protocol (AMP) 2015
  - Submitted as set of IDs to DTNWG.

# How Do We Manage Networks Today? Do we need a new thing?

Low-latency approaches to network management fail to scale with increasing delays and disruptions.

- Rich set of evolving capabilities
  - Simple Network Management Protocol (SNMP).
    - Pull model of information from managed devices.
      - > Support for "traps" to push unreliable notifications of pre-defined events.
  - Network Configuration Protocol (NETCONF).
    - XML-based, session-based remote-procedure call (RPC) interface for node configuration.
  - Remote Network Monitoring MIB (RMON).
    - Mechanisms for exchanging network monitoring data.
- Poor scaling with delays, disruptions, or commanding
  - Focus on getting data to operators.
  - Less focus on in-situ response options.
  - Reliance on scripting and mobile code which is not always a deployment option.

New capabilities have come online such as RESTConf, Yang 1.1, and YangPush.

These capabilities add more "push" semantics, but still struggle to operate in highly disrupted scenarios. <u>The AMA</u> <u>is being reworked</u> to focus more effectively on the unique characteristics of challenged networks.

### **Asynchronous Management Architecture**

Let's think through an extreme use case...



### Intersection: Spacecraft and Data Center?

# How do we manage partitioned networks, such as:

- New Space Networks
- Challenged Sensor Networks
- Vehicle Networks
- Resource-Constrained Networks
- Disruption-Tolerant Networks

# Spacecraft have this problem, what do they do?

- Autonomy models
- VERY efficient encodings
- Low requirements for processing.

# Can we bring these worlds together in an IETF-useful way?

Spacecraft Fault Management Systems

- Stim/Resp Systems
- Heritage Implementations
- Deterministic Processing
- Mission Specific Tools
- Less Infrastructure Funding
- Network CONOPS

#### <u>Terrestrial Network</u> <u>Management - Datacenters</u>

- Lots of Standards
- Open-source tools
- Large Investments
- Immature autonomy
- Inefficient implementations
- Inefficient protocol layering

#### What does a healthy intersection of these worlds look like?

- Automation and Autonomy Model
- Network Management Standards
- Compatibility with commercial work

### **AMA/ADM/AMP Interactions**

#### Asynchronous Management is network management via configured, deterministic autonomy.



It is defined as:

(3) A Protocol

- (1) An Architecture https://datatracker.ietf.org/doc/html/draft-ietf-dtn-ama-03
- (2) A Data Model https://tools.ietf.org/html/draft-birrane-dtn-adm-03
  - https://datatracker.ietf.org/doc/html/draft-birrane-dtn-amp-08

# Work is ongoing in the DTN-WG



# **Active Drafts (For Reference)**

- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-ama/</u>
- https://datatracker.ietf.org/doc/draft-birrane-dtn-amp/
- https://datatracker.ietf.org/doc/draft-birrane-dtn-adm/
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-agent/</u>
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-bp/</u>
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-bpsec/</u>
- https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-ion-bpadmin/
- https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-ion-ipnadmin/
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-ion-ltpadmin/</u>
- https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-ionadmin/
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-ionsec/</u>
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-adm-ltp/</u>
- <u>https://datatracker.ietf.org/doc/draft-birrane-dtn-ampmgr-sql/</u>

## **AMA: Overview**

From https://datatracker.ietf.org/doc/html/draft-ietf-dtn-ama-03

- Service Definitions
  - Configuration: Change settings on an Agent.
  - Reporting: Receive performance information from an Agent.
  - Autonomous Parameterized Control: Change Agent Behavior.
  - Administration: Fine-grained access to abilities.
- Desirable Properties
  - Intelligent Information Push: Can't rely on others.
  - Minimize Message Size: Increase probability of delivery.
  - Absolute Data Identification: pre-shared, global naming when possible.
  - Custom Data Definition: Only send minimal necessary data sets.
  - Autonomous Operation: Decisions local to Agent based on its config.

# **AMA: The Simple System Model**

- Agents
  - Run on Managed Devices
  - Configure/Report on devices
  - Heavy autonomy and parameterized control
- Manager(s)
  - Collect/Fuse data from Agents
  - Configure Agent behavior
  - Open-loop control
- ADMs
  - Well-named Data and Controls
  - Superset of MIB
  - Move to describe them in YANG
  - Preconfiguration reduces msg size



# **ADM Template: Logical Modeling**

- · Separate the data specification from its encoding.
  - Use AMP specification to define how to compactly encode ADM items
- ADMs Schemas will define logical models
  - Designed to identify minimum set of information per data model
  - Remove any "encoding hints" from the models.
  - Use the YANG modelling language
    - Tools exist to validate YANG schemas for correctness and plot dependencies.

#### • ADMs will be defined in JSON

- Conventions will be defined to make JSON writing expressive and "easy"
- Reuse existing notations/delimiters where possible (query string)
- Define compilers/adapters
  - Presuppose adapters/compilers to generate encodings as necessary

# **Asynchronous Management Model (AMM)**

#### "Atomic" Elements

- EDDs: collected by agents.
- Literals: useful constants.
- **Ops**: opcodes for math functions.
- Ctrls: opcodes for agent behavior.
- Tables: Structured data sets
- "Dynamic" Elements
  - Vars: strong-typed variables
  - Macro: Ordered set of Ctrls.
  - Rpts: Ordered sets of data
  - Rules: Time or State based autonomy.

### An AMM defines 9 types of data for each application/protocol managed in the AMA.



#### **BP Example: Externally Defined Data**

```
"edd": [
     "name": "num_pend_reassembly",
     "type": "UINT",
     "description": "number of bundles pending reassembly"
    },
     "name": "bundles_by_priority",
     "type": "UINT",
     "parmspec": [{"type":"UINT", "name":"mask"}],
     "description": "Number of bundles for the given priority. Priority is given as a priority mask where
                     Bulk=0x1, normal=0x2, express=0x4. Any bundles matching any of the masked
                    priorities will be included in the returned count"
    },
```

#### **BPSec Example: Tables**

```
"table-templates" : [
    {
        "name": "ciphersuites",
        "columns": [{"type":"STR", "name":"csname"}],
        "description": "This table lists supported ciphersuites."
    },
    {
        "name": "bib_rules",
    }
}
```

```
"columns": [{"type":"STR", "name":"SrcEid"},
        {"type":"STR", "name":"DestEid"},
        {"type":"UINT", "name":"TgtBlk"},
        {"type":"STR", "name":"csName"},
        {"type":"STR", "name":"keyName"}
],
"description": "BIB Rules."
```

```
APL
```

},

### **IonAdmin Example: Controls**

```
"controls": [
   "name": "node_contact_add",
   "parmspec": [
    {"type":"TS",
                         "name":"start"},
    {"type":"TS",
                         "name":"stop"},
    {"type":"UINT",
                          "name":"node_id"},
                         "name":"dest"},
    {"type":"STR",
                         "name":"data_rate"},
    {"type":"FLOAT32",
    {"type":"FLOAT32",
                         "name":"prob"}
   ],
```

"description": "This control schedules a period of data transmission from node\_id to dest. The period of transmission will begin at start and end at stop, and the rate of data transmission will be data\_rate bytes/second. Our confidence in the contact defaults to 1.0, indicating that the contact is scheduled - not that non-occurrence of the contact is impossible, just that occurrence of the contact is planned and scheduled rather than merely imputed from past node behavior. In the latter case, confidence indicates our estimation of the likelihood of this potential contact."

### **Asynchronous Management Protocol**

A CBOR encoding of ADM information.



Format	1	[1	, [3	2, 3	3]]					I [.	. 1,	[2	, 3	]]			
RFC 713	1	c2	05	81	cZ	02	82	83		+							
	1									1							
ASN.1 BER	F	30	Øb	02	01	01	30	06	02	1 30	80	02	01	01	30	06	02
	I	01	02	02	01	03				1 0:	02	02	01	03	00	00	
	1									1							
MessagePack	1	92	01	92	02	03				1							
	1									1							
BSON	T	22	00	00	00	10	30	00	01	1							
	1	00	00	00	04	31	00	13	00	1							
	T	00	00	10	30	00	02	00	00	1							
	L	00	10	31	00	03	00	00	00	1							
	1	00	00							1							
	1									1							
UBJSON	L	61	Ø2	42	01	61	02	42	Ø2	1 6:	L ff	42	01	61	02	42	02
	I	42	03	ź.						42	2 03	45					
CBOR		82	01	82	02	03				1 9	F 01	82	02	03	ff		

#### AMP is:

- Independent of BP
  - AMP messages can use any transport.
- A binary encoding of ADM Information
  - CBOR is used because it is compact and well supported.
  - CBOR encoders can be < 10KB in size.
- A set of messages to exchange these encodings
  - Simple messaging, like SNMP and NETCONF
    - Register Agent
    - Report Set
    - Perform Control
  - Most AMP Agent control is captured in an AMP Agent ADM.
- A state machine
  - Describing the behavior of agents and managers in a network.

This is a content encoding, and not necessarily a transport encoding.

AHF	nessage Group ro	Jimac
	+	-+
	Message Group	
	[ARRAY]	i i
	+++	+
	11	
	ii ii	
/		
++	+	++
Timestamp	Message 1	Message N
[TS] [	(BYTESTR)	[BYTESTR]
	[	+

Figure 23

#### **ION Reference Implementation**

AMP Agent, AMP Text-Based Manager

- AMP in ION Release 4.0
  - Updated directory structure
  - Support for BPv6 and BPv7 -
  - Feature complete to latest AMP spec
    - https://tools.ietf.org/html/draft-birrane-dtn-amp-08



adm

#### What's New?

\_

- New directory design C nm\_agent C)nm\_mg This diagram is an idealized concept of NM-related dependencies (not as implemented ADMs can be stored with their applications ADMs Agent Linked as needed to agent/mgr Clibion C libnmMgrAdm C libnmAgentADM Future: Theoretically replaceable with alternate implementations Endian-Neutral CBOR qcbor Shared - Test Support C other\_adms\_impl C libnm C bp\_adm C bp\_adm\_impl C other\_adms Additional regression tests C other\_adm\_deps C) qcbor (C) libbp

# **AMP-ION Agent**



#### **AMP-ION Manager**



#### **ADM Auto Generation**

#### • ION C-generating AMP Python Script (CampPython)

- Github project
  - CAPI for AMP defined for ION (3.x or 4.x line)
  - Produces .c/.h files per ADM
  - Includes a user-customizable .c/.h) and "round-tripping"
  - camp <adm.json> -c <old\_impl.c> -s <old\_impl.h>
- Adding and maintaining ADMs much simplified
  - But, now, lots of data. 200+ data items, ~100 controls/operators







#### **CAMP User Implementation File Example**

value t adm bpsec get ciphersuite names(tdc t params)

```
value t result;
* |START CUSTOM FUNCTION get ciphersuite names BODY
* +------+/
   char *tmp = bpsec instr get csnames();
   result.value.as ptr = STAKE(size));
   memcpy(result.value.as ptr, tmp, size);
   SRELEASE(tmp);
   result.type = AMP TYPE STRING;
* |STOP CUSTOM FUNCTION get ciphersuite names BODY
return result;
```

# **AMP Manager SQL Schema**

#### AMP Schema

- Abstracts any binary encoding from user.
- SQL-enabled AMP Manager:
  - Polls DB for new info to send to AMP Agents.
  - Stores received agent information for visualization.

#### AMP-Specific Tables

- Data representations for constants, literals
- Static ADM Information, Operator-defined data
- Outgoing/Incoming message groups
- CAMP integration
  - CAMP generates SQL files from JSON ADMs
- Testing
  - SQL scripts to auto-populate tables for testing



#### Master's Thesis: NETCONF <-> AMP

#### Jean Chorin

- Visiting UMBC from Dresden (works for Marius)
- NETCONF is "the" current NM protocol
  - YANG is its modelling language
- NETCONF isn't a DTN solution
  - Requires sessions and state. No Autonomy. XML encoding. Can't send delta updates. Chatty, closed-loop control CONOPS.
  - No S/C is going to run a NETCONF agent.
- ADM <-> YANG converter
  - Model mapping. Scripts to populate NETCONF stores from DTN.
  - Handles asynchronous-to-synchronous data issues.

### **External Work: Integrate with NETCONF**



#### **External Work: Integrate with NETCONF**



#### Live Demo from IETF 104

#### https://www.youtube.com/watch?v=-AT7mF8Gn94

Activities 🗂 Terminal +		in the second	Max Mar 25, 65:49			h- CA#0-
				CORE (42774 on utrunts) sing	pla hop into	000.0
and the second s	rootOnt: /imp/pycers.42774/n1.conf	000	e Edit Canvus View Josis Widgets Session	9.0		
PHE tdlt View Search Termin CC CCbecking Server Modules CC CCWarning: skipping enterpr Warning: skipping enterpr Warning	<pre># Help</pre>	stywngitet (ywrai23-s	Ja Saz 124	eth1 0.2 14 mJ 10.3.3.3/74 J		Script für 30-30 acen 10-30 acen 11 • resultion m
<pre>yangcl1 adming10.3.3.5&gt; a action count yangcl1 adming10.3.3.5&gt; a File EDE View Seatch Terms rest[0.3]/tmp/pycore.4277/ 360 rest[0.3]/tmp/pycore.4277/ 360 rest[0.3]/tmp/pycore.4277/ mts/0/bjlomarbett//lmplemer 2019-03-25 05:49:15,000 2019-03-25 05:49:15,000 2019-03-25 05:49:15,000 2019-03-25 05:49:15,000 2019-03-25 05:49:15,000 2019-03-25 05:49:15,000 2019-03-25 05:49:15,000 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201 2019-03-25 05:49:15,201</pre>	64_thr description id period start def_thr could resident (imployments) and reso (main_f) (imployments) (main_f) (imployments) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (paramike.transport) (imployments) (passed) (controller.mg.controller) (imployments) (passed) (controller.mg.controller) (imployments) (passed) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (passed) (controller.mg.controller) (imployments) (controller.mg.controller) (imployments) (controller.mg.controlle	Starting 'nn_ngr' program - Megistering all agents Registering agent at EID ( 2.0, citent libssh2_1.8.0) ), ord).	dport 830 -3 REDIRECTto-port 8 s/server-key -n /hone/core/Docune with arguments: ipn:3.6 pn:4.5	Tot View Bearch Terminal and Mai/ <u>thp/pyCore.42774/M4.co</u> hared/utlis/db.ci19] : db hared/utlis/db.ci559] : vd hared/utlis/db.ci565] : vd hared/utlis/db.ci565] : vd hared/utlis/db.ci565] : vd hared/utlis/db.ci563] : vd	nt Amplyment 427 nf# nm_agent ipr ,read_objs Creat B_init Added 0 4 B_init Added 0 8 B_init Added 0 8 B_init Added 0 8 t_nain Thread 0 4 if_send Sending	74/44.com 14.5 ipn:3.6 ing OD: magent.db ontrols from OB. eport Template Definitions from D wile Definitions from DB. ariable Definitions from DB. tarted B#200499800709780e3#342e35 to ip

APL

#### amp.me – URI Decoding

$\leftrightarrow$ $\rightarrow$ C $\odot$ localhost:	3000/#amp.me		k 🛛 🔾								
¥ Mgr ∅ + Ĉ   ✔ DB ∅	> * C										
AGENTS	Currently accepts CBOR Hex (0x) AMP-CBOR (ie: ari:0x), or AMP URIs (ie: ari:/IANA:). This tool is experimental and some data types may not be fully implemented at this time.										
📄 All	Input (CBOR or URI) * ari:/IANA:amp.agent/Ctrl.gen_pts[[ari:/IANA:amp.agent/Rptt.full_report()].[])										
🗙 ipn:3.64											
AMP DB	Parse Clear										
Received Sets	Trace Summary CBOR Encoding UML										
Controls	Input: ari:/IANA:amp.agent/Ctrl.gen_rpts([ari:/IANA:amp.agent/Rptt.full_report()].[])										
sample	Parsed As: ARI										
NM MANAGER	CBOR: c11541050502252381871819410000										
Status	URI										
Send Command	ari1/IANA:anp.agent/Ctrl.gen_rpts([ari1/IANA:amp.agent/Rptt.full_report()],[])										
UTILITIES	ISON										
ADM Listing	( """"""""""""""""""""""""""""""""""""										
Decumentation	"type": 1, "namespace": 1,										
Support	"adm_type": 1, "parameters": (										
	"class": "TWVC", "values": (										
	<pre> 'class': 'AC',  'ari': (</pre>	Trace Summary CBOR Encod	ling UML								
	"class": 'ARI", "type": 7, "namespace": 1,	Input	Туре	Value							
	"adn_type": 5 }		ARI	ari:/0x8718194100							
	), { folase": "TRVC".		AC	ac:/0x818718194100							
	"values": [] }		TNVC	tnvc:/0x0502252381871819410000							
	}, "name": 5		ARI	ari:/0xc11541050502252381871819410000							

#### amp.me – CBOR Decoding & Re-encoding

nput	(CBOR or URI	I) ari:/0xc11541	0505022523818	71819410	0000		Trace	Summary	CBOR Enco	oding	UML				
							Input	,		Туре			Value		
			Parse	Clear						ARI			ari:/0x87	718194100	
	1									AC			ac:/0x81	8718194100	
										TNVC			tnvc:/0x	050225238187181941	0000
										ARI			ari:/0xc1	15410505022523818	71819410000
			~												1
	Trace	Summary	CBOR Encodi	ng	UML										
	Input			Туре		Value									
	c1			AMP_TYPE	.CTRL	NN=true,PARM=tru	ue,ISS=false,TA	G=false							
	15		4	ARI.Nickna	ame	NN=21,NS=1,TYPE	=1				Trace	s Su	mmary	CBOR Encoding	UML
	4105			ARI.name		[object DataView]									
	05			TNVC.flag		5		€				ARI			
	2			TNVC.leng	th	2		1				AM	P_TYPE.0	CTRL - 1	
	25			TNVC.type	2	AMP_TYPE.AC									
	23			TNVC.type	2	AMP_TYPE.TNVC								$\backslash$	
	81			AC.start		1								$\langle \rangle$	
	87			AMP_TYPE	RPTTPL	NN=true,PARM=fal	se,ISS=false,TA	G=false					<u><u> </u></u>	$\langle \rangle$	
	1819			ARI.Nickna	ame	NN=25,NS=1,TYPE	=5				Nick	kname			
	4100			ARI.name		[object DataView]					ТО	DO		Name	
	00			TNVC.flag		0					0x <sup>-</sup>	15 = 21	- 11	05	
				ARI		ari:/0x8718194100	)				NS Tur	5=1 00-СТРІ	_1		
				AC		ac:/0x8187181941	00				I I I		-' /		
				TNVC		tnvc:/0x05022523	8187181941000	0							
				ARI		ari:/0xc115410505	0225238187181	19410000							

#### **SQL** Insertion

```
C02YD01EJG5J-ML:nm_ui edelldj1$ ./sql_msgs_insert.js --msg "ari:/IANA:amp.agent/Ctrl.gen_rpts([ari:/IANA:amp.agent/Rptt.full_rep
ort()],[])" --agent "ipn:3.64"
undefined
[Object: null prototype] {
  ns: 'IANA:amp.agent',
  namespace: 'IANA',
  nsname: 'amp.agent',
  cmd_type: 'Ctrl',
  cmd_name: 'gen_rpts',
  args: '[ari:/IANA:amp.agent/Rptt.full_report()],[]'
tnvc fromuri part= [ari:/IANA:amp.agent/Rptt.full_report()] from uri= [ '[ari:/IANA:amp.agent/Rptt.full_report()]', '[]' ] typ
e = AC
[Object: null prototype] {
  ns: 'IANA: amp.agent',
  namespace: 'IANA',
  nsname: 'amp.agent'.
  cmd_type: 'Rptt',
  cmd_name: 'full_report',
  args:
tnvc fromuri part= [] from uri= [ '[ari:/IANA:amp.agent/Rptt.full_report()]', '[]' ] type= TNVC
insertCtrl( 112, null, ari:/IANA:amp.agent/Rptt.full_report()
insertCtrl( 118, 24, ari:/IANA:amp.agent/Ctrl.gen_rpts([ari:/IANA:amp.agent/Rptt.full_report()],[])
Created message group with id of 1
Done
```

#### **ADMs Listing**

APL

- This listing is generated by parsing the ADM JSON files
- Future enhancements would allow viewing details per ADM from the JSON files and/or provide an equivalent view from the MySQL database.

AGENTS	Name	Index	IETF Source	Issuer	Controls	Edds	Mdats	Rptts	Tbits	Vars	Consts	Operators
📄 Ali	amp.agent	1	https://tools.ietf.org/html/draft-birrane-dtn-adm-agent-05	IANA	16	14	4	1	6	1	1	53
¥ inn:3.64	bp.agent	2	https://tools.ietf.org/html/draft-birrane-dtn-adm-bp-03	IANA	1	29	4	2	0	0	0	0
A 1911.3.01	ltp.agent	3	https://tools.ietf.org/html/draft-birrane-dtn-adm-ltp-01	IANA	1	42	4	1	1	0	0	0
AMP DB	bpsec	4	https://tools.ietf.org/html/draft-birrane-dtn-adm-bpsec-01	IANA	10	55	4	2	4	1	0	0
Outgoing Sets	bpadmin	5	https://tools.ietf.org/html/draft-birrane-dtn-adm-ion-bpadmin-01	IANA	26	1	4	0	6	0	0	0
Received Sets	ipnadmin	6	https://tools.ietf.org/html/draft-birrane-dtn-adm-ion-ipnadmin-00	IANA	6	1	4	0	2	0	0	0
	ion.admin	7	https://tools.ietf.org/html/draft-birrane-dtn-adm-ionadmin-01	IANA	15	14	4	0	2	0	0	0
Controls	ionsec	8	https://tools.ietf.org/html/draft-birrane-dtn-adm-ionsec-01	IANA	12	0	4	0	2	0	0	0
sample	ltp.admin	9	https://tools.ietf.org/html/draft-birrane-dtn-adm-ion-ltpadmin-01	IANA	9	1	4	0	1	0	0	0
NM MANAGER	Summary				96	157	36	6	24	2	1	53
Status												
Send Command												
UTTLITTES												
Ondries												
Amp.me												
ADM Listing												
9												

https://youtu.be/dNvIdiZkIWY?t=7078

IETF1 DTN	09				
Contrast Concern	CONTRACTOR NAMES IN CONTRACTOR	100 ten 100		R	
C sur market	(nemas) + 🖉 (of lating frames)	*.(.*.			
AN RICKS	2				111 A
45475	Committe acceste CRCR	Data line 1 4000 CMCR (her art line 1, or ANN 180	to be an Africa. 1 This had a second second and some	a first large state and he had a measurement of this large	and the second second
	has state a ship at this are april Of	are under ANA to apertRet M. wort(3.0	a de maine de las de		and the second se
w 1013.54					
			. Para		
Congoing Same	Trea Survey COB Insuing	(m)			
Committee	Sheet.	Turn	(Mar		
IZ-man-					
on American					
Baid Carmand					ALL
-same .				and the second se	
Arts.com				the state of the state of the state	
Distances and				the second s	States and the second
C Bargers				and the second se	
					SHE Y
					+
Image:	1:57:58 / 2:08:38	and residence per a liter		~	■ 🗢 🖬 🗗 🖸

APL



#### JOHNS HOPKINS APPLIED PHYSICS LABORATORY