

CGNS Steering Committee Telecon Minutes
16 July 2013

- 1) The telecon was called to order at 11:00 AM eastern time. There were 9 attendees, listed in Attachment 1.
- 2) The minutes of the 30 April 2013 telecon were approved.
- 3) Rumsey described a draft document to help users go from v2.5 to v3.x. Main comment from the committee is that a better introduction is needed. A question was raised whether there is still a problem using partial reads and writes with HDF5. Documentation of the original problem could not be found, but nonetheless Wedan confirmed that it is not a problem in the current software. Wedan pointed out changes_from_2.5.txt and changes_from_3.0.html files that come with the distribution. Rumsey to finish document to help users go from v2.5 to v3.x and will post it to Downloads page.
- 4) Discussion on connectivity
 - a) Issue of multizone unstructured files was discussed. Arunajatesan and Howard of Sandia described the issue. Main problem is that many grid generator codes are not writing enough information to make CGNS multizone unstructured files easy to use. CGNS provides the capability, but many of the relevant nodes are optional. Should the nodes be required? Arunajatesan and Howard will provide examples to the committee, to demonstrate the problem as well as desired solutions. The committee was leaning toward not making any new requirements (i.e., keeping nodes optional), but rather posting a usage document of suggested practices, to help users and software vendors write such files. Arunajatesan and Howard to provide list/examples for multizone unstructured CGNS usage.
 - b) A somewhat related issue exists when writing a CGNS file with both structured and unstructured zones. This is currently possible with existing MLL, because the grid type of the donor zone is known, but there may be an issue in the SIDS that needs to be fixed. Wedan to investigate needs related to combining structured and unstructured zones in a CGNS file.
- 5) V3.2: still in beta status. Wedan recently released updates. The latest allows user to set different MPI communicators for different groups, and made fixes to cmake scripts. Wedan says v3.2 it is basically ready to come off beta, but we still need committee members to test it, especially the parallel capability. Committee will wait until next telecon to make a decision regarding taking v3.2 off of beta.
- 6) Rumsey described status of updating the AIAA Recommended Practice document. Steering Committee members have been asked to send a sign-up form in to Amy Barrett of AIAA. After we get a balanced subcommittee, we must review the SIDS v3.2(beta) and approve it as the latest update, before AIAA will publish it.

- 7) Informal CGNS meeting was held in San Diego. There, we discussed issue of multi-zone connectivity for unstructured grids (see item 4 above), and also discussed Marc Poinot's document regarding compliance (Attachment A).
- 8) Status of Old Action Items:
 - a) Hauser and Duque to continue to look into the consortium idea for CGNS, including more active support of HDF-5 consortium. Also look into applying to NSF software infrastructure for sustained innovation, possibly related to data management plan.
 - i) Carries.
 - b) Hauser, Duque, and Iannetti to continue to develop Iannetti's proposal for handling sprays of unconnected points.
 - i) Carries.
 - c) Hauser to finalize the CGNS survey based on feedback from committee members, and email out to CGNSTalk.
 - i) Carries.
 - d) Guzik will summarize the changes required to the MLL software to have core data always start at location (1,1,1) when including Rind data, as well as backward-compatibility implications, to help the Steering Committee decide on a course of action (index discrepancy).
 - i) Carries.
 - e) Rumsey will send out a reminder to the committee to vote on the SIDS as AIAA recommended practice, as well as testing of V3.2.
 - i) Done.
- 9) Next Telecon is tentatively set for Wed, Sept 18, 2013, 11 am eastern. A confirmation email will be sent prior to the meeting.

10) Summary of **action items**:

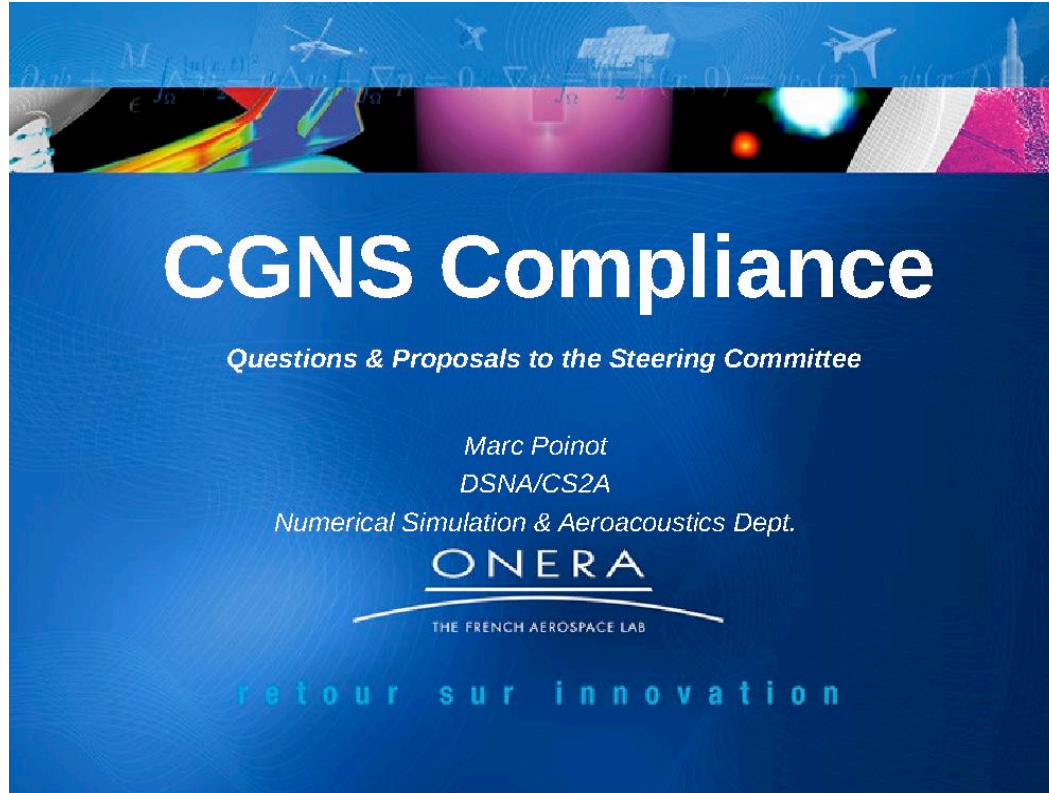
- a) Hauser and Duque to continue to look into the consortium idea for CGNS, including more active support of HDF-5 consortium. Also look into applying to NSF software infrastructure for sustained innovation, possibly related to data management plan.
- b) Hauser, Duque, and Iannetti to continue to develop Iannetti's proposal for handling sprays of unconnected points.
- c) Hauser to finalize the CGNS survey based on feedback from committee members, and email out to CGNSTalk.
- d) Guzik will summarize the changes required to the MLL software to have core data always start at location (1,1,1) when including Rind data, as well as backward-compatibility implications, to help the Steering Committee decide on a course of action (index discrepancy).
- e) Rumsey to finish document to help users go from v2.5 to v3.x and will post it to Downloads page.
- f) Arunajatesan and Howard to provide list/examples for multizone unstructured CGNS usage.

- g) Wedan to investigate needs related to combining structured and unstructured zones in a CGNS file.

Attachment 1: Attendees

Srinivasan Arunajatesan
Pat Baker
Bob Bush
Thomas Hauser
Dan Hiepler
Micah Howard
Dmitri Kamenetskiy
Chris Rumsey
Bruce Wedan

Sandia Nat Lab.
Pointwise
Pratt & Whitney
University of Colorado
Intelligent Light
Sandia Nat Lab.
Boeing
NASA Langley
Computational Engineering Solutions



The slide features a blue background with a collage of images at the top, including mathematical equations, aircraft, and aerodynamic flow visualizations. The main title 'CGNS Compliance' is in large white font. Below it, the subtitle 'Questions & Proposals to the Steering Committee' is in a smaller white font. The presenter's name 'Marc Poinot' and affiliation 'DSNA/CS2A' are listed, followed by the department 'Numerical Simulation & Aeroacoustics Dept.'. The ONERA logo is centered, with the tagline 'THE FRENCH AEROSPACE LAB' and the slogan 'retour sur innovation' at the bottom.

CGNS Compliance

Questions & Proposals to the Steering Committee

Marc Poinot
DSNA/CS2A
Numerical Simulation & Aeroacoustics Dept.

ONERA
THE FRENCH AEROSPACE LAB

retour sur innovation

Motivation: probably more questions than answers!

- ▶ Many questions...
 - ▶ You are CGNS compliant if you use CGNS/MLL
 - How to find a bug in cgnscheck?
 - Both CAD tool and CFD solver pass cgnscheck, but they cannot interoperate?
 - How can I produce a non-compliant transient file in a complex workflow?
 - ▶ It is not clear if a tool has to fail or to ignore
 - What if the CGNS tool reads the Mach but uses another one?
 - What if your solver doesn't know about HEXA_56, DataClass_t?
 - ▶ Should we avoid extension because of backward compatibility?
 - How to add a reference frame for grids?
 - How to allow name referencing between different CGNSBase_t?
 - How to define a cartesian grid?
 - ▶ Now we are going to multi-physics 'open' workflows
 - How to check interoperability off-line & stand alone?
 - How to extend grammar to CAA, CSM, FM...?
 - How to avoid to have a 'stuff & misc other stuff' storage?
 - ▶ Would CGNS help me to reduce computation cost?
 - Is a compliant CGNS file a guarantee my computation won't crash?
 - Is it actually possible to blindly run a CGNS compliant computation?
 - ▶ And so on...

Status

- ▶ Today validation is made through CGNS/MLL
 - ▶ try a cg_open
 - ▶ pass a cgnscheck
 - ▶ use any other tool reading/writing CGNS/HDF5
- ▶ pyCGNS based tools at Onera
 - ▶ Develop for Onera needs using CGNS/Python
 - Capability to generate transient non-compliant in workflow
 - ▶ A compliance checking tool CGNS.VAL
 - Embedded into applications + stand alone tool + CGNS.NAV GUI
 - Comes with separate SIDS and elsA test suite: pass/fail files
 - Grammar specialization by OO mean: elsA is at least SIDS
 - uses CHLone instead of CGNS/MLL
- ▶ Obviously, certification of 'something' already exists
 - ▶ Then we may find somebody, somewhere, who knows about!
 - ▶ Our system is close to DBMS & compiler validation

Glossary

- ▶ **Compliance (1):** ability to follow all guidelines described into the standard
- ▶ **Compliance (2):** ability to successfully pass the certification process
- ▶ **Process:** suite of input/action/output
- ▶ **Standard:** defines the contents (document)
- ▶ **Certification:** process to check actual contents w.r.t. standard
- ▶ **Label:** result of the certification as a stamp on target tool
- ▶ **Level:** of certification and/or label, a subset of required topics to achieve complete certification
 - But, formally speaking, you are compliant or you are not!
 - Introducing levels helps to support volunteers
- ▶ **Target Tool,** Target Tool set: one or more software taken as a whole for a certification
- ▶ **Pass tree:** a CGNS tree that should pass the check without error/warning
- ▶ **Fail tree:** a CGNS tree that should raise **one** expected error/warning

CGNS compliance targets

► Main components



CGNS/SIDS

- Defines a data model for CFD
- Product is a document describing a more or less formal grammar



CGNS/CGIO

- Defines the mapping from CGNS/SIDS to actual implementation
- Product is a document describing data contents and software system requirements/ limits



CGNS/MLL

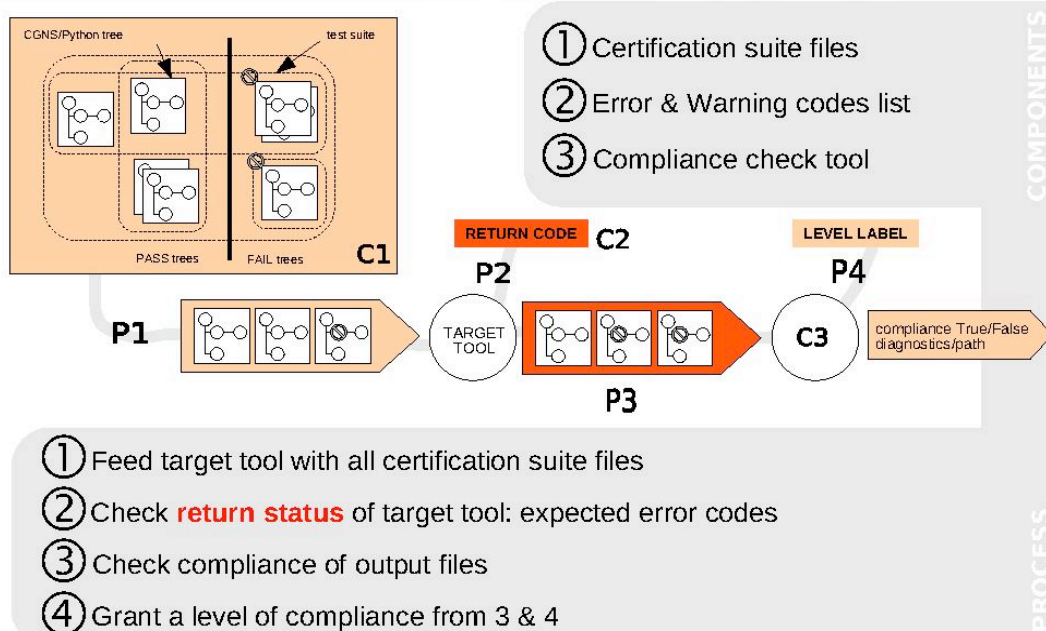
- Implementation of CGNS/HDF5 mapping, provides the user with an API
- Product is a software library to be embedded into C/C++/Fortran, also has basic tools built with this library



CGNS/HDF5

- Actual file containing a mapping of CGNS/MLL as HDF5 (or ADF)

Compliance process proposal



Check tool

- ▶ A check tool would...
 - ▶ Pass on all pass files & Fail on all fail files
 - ▶ &Give the node path with its error code
 - ▶ Detect CGNS/SIDS version
 - ▶ Accept CPEXs
 - ▶ Maybe the good check tool is an Open Source tool that allow any user to check compliance by himself
 - ▶ And would accept grammar specialization for users' own purpose
 - ▶ Any tool that succeed in passing the certification suite can be seen as a potential compliance checking tool
-
- ▶ Demo with CGNS.VAL

NSCOPE-PRS-020/7/10

M.Poinot DSNA/CS2A



Error codes examples

```
[G001:E] CGNSLibraryVersion [%s] is too old wrt check level
[G002:E] CGNSLibraryVersion is incorrect
[G003:E] Name [%s] is not valid
[G004:E] Name [%s] is a duplicated child name
[G005:E] PANIC: Cannot find node with path [%s]
[G006:E] PANIC: Node data is not numpy.ndarray or None
[G007:E] PANIC: Node children is not a list
[G008:E] PANIC: Node name is not a string
[G009:E] PANIC: Node is not a list of 4 objects
[G010:E] PANIC: Node is empty list or None
[G011:E] PANIC: Node name is empty string
[G012:E] PANIC: Node name has forbidden chars
[G013:E] PANIC: Node name is . or ..
[G014:E] PANIC: Node name is too long
[G015:E] Bad node value data type

[S001:E] Unknown SIDS type [%s]
[S002:E] SIDS type [%s] not allowed as child of [%s]
[S003:E] SIDS type [%s] not allowed for this node
[S004:E] DataType [%s] not allowed for this node
[S005:E] Node [%s] of type [%s] not allowed as child
[S006:E] Node [%s] of type [%s] allowed only once as child
[S007:E] Node [%s] of type [%s] is mandatory
[S008:E] Child name [%s] reserved for a type in [%s]
[S009:E] Bad node shape [%s]
[S010:E] Bad node value

[S101:E] Unknown ZoneType value
[S102:E] Unknown SimulationType value
[S103:E] Unknown GridLocation value
[S104:E] Unknown GridConnectivityType value
[S105:E] Unknown DataClass value

[S120:E] Name [%s] is reserved for a child of type [%s]
[S121:W] Name [%s] is not known as a CGNS/SIDS identifier
[S150:W] No default DataClass set
[S151:W] Default GridLocation is set to Vertex
[S152:W] Default GridConnectivityType is set to Overset
[S153:W] DataClass defined without DimensionalUnits
[S154:W] Local DataClass refers to a parent DimensionalUnits
[S155:W] DataArray refers to a parent DataClass
[S156:E] DimensionalExponents without DataClass
[S157:E] DimensionalExponents without DimensionalUnits
[S158:W] DataConversion without DataClass
[S159:W] DataConversion without DimensionalUnits
[S160:W] DataConversion without DimensionalExponents
[S161:E] DataClass requires DimensionalUnits

[S191:E] Bad node value shape
[S192:E] Unexpected shape %s instead of %s for [%s]
[S193:E] Missing required node of type [%s] in [%s]
[S194:E] Missing required node of name [%s]
[S195:E] Bad child node value shape [%s]
[S196:E] Bad child node value datatype [%s]
[S197:E] Inconsistent child node value [%s]

[S201:E] Inconsistent PhysicalDimension/CellDimension
[S202:E] Bad value for CellDimension
[S203:E] Bad value for PhysicalDimension
[S204:E] Bad Transform values
[S205:E] Bad ElementSizeBoundary value
[S206:E] PointRange values out of range
[S207:W] PointRange values not ordered
[S208:E] PointRangeDonor values out of range
[S209:W] PointRangeDonor values not ordered
[S220:W] BCTYPE is Null
[S221:W] BCTYPE is UserDefined
[S260:E] GridCoordinates node has no coordinate array
[S261:E] Coordinates number different from PhysicalDimension
[S262:E] Size of coordinates different from zone VertexSize
[S263:E] Unable to identify coordinate system from names
[S264:E] Coordinate name not allowed for 1D system
[S265:E] Coordinate name not allowed for 2D system
[S266:E] Coordinate name not allowed for 3D system
[S270:W] Descriptor contains non-printable or non-ascii chars
[S280:E] DiffusionModel should have a value
[S281:E] DiffusionModel value should have dimension of %d
[S282:E] DiffusionModel value should contain only 0s or 1s

[S300:E] FamilyName is empty
[S301:E] Reference to unknown family [%s]
[S302:E] Reference to unknown additional family [%s]
[S303:W] Family is never referenced

[S400:E] Cannot get connectivity donor zone
[S401:E] Connectivity donor zone [%s] not found
[S501:E] Reference to unknown node [%s] in [%s]

[S502:E] Bad iteration number [%s]
[S503:E] Name [%s] is reserved for time-dependant pointers
[S504:E] Name [%s] is reserved for ConvergenceHistory

[S600:W] CGNSTree has no CGNSBase
[S601:W] CGNSBase has no Zone
[S602:E] Zone has no GridCoordinates
[S603:E] No GridCoordinates_t of name GridCoordinates in zone
[S604:W] ZoneBC has no BC
```

NSCOPE-PRS-020/8/10

M.Poinot DSNA/CS2A



Tentative levels & labels

- ★
 - ▶ Read/write a compliant CGNS/HDF5 file
- ★★
 - ▶ 1+
 - ▶ Return error codes
- ★★★
 - ▶ 2+
 - ▶ Warning if ignored data found
- ★★★★
 - ▶ 3+
 - ▶ Take into account all data found in tree
- ★★★★★
 - ▶ 4+
 - ▶ Suggests correction/modification to input trees



what about... links, parallel, multi-thread, CGNS/MLL use, userdefined data use, scoped data masking, both ADF/HDF, CGNS/Python interface, embedded files, lazy load/save...

CGNS Steering Committee actions

- ▶ Define error/warning codes
 - ▶ CGNS/SIDS codes
 - ▶ mapping implementation codes
- ▶ Define levels of compliance
- ▶ Define/create/maintain certification suite files
 - ▶ Pass files raise no error
 - ▶ Fail files raise one expected error/warning code
- ▶ Define/create/maintain compliance checking tool

This is too big effort
regarding actual people producing for CGNS today

If we have no mean to check CGNS/SIDS compliance,
the standardisation would **fail**