

Building Block macro principle for Standard Program Library Design

Phuse Brighton

October 2011

Francis Dsa

frd@novonordisk.com

Novo Nordisk at a glance

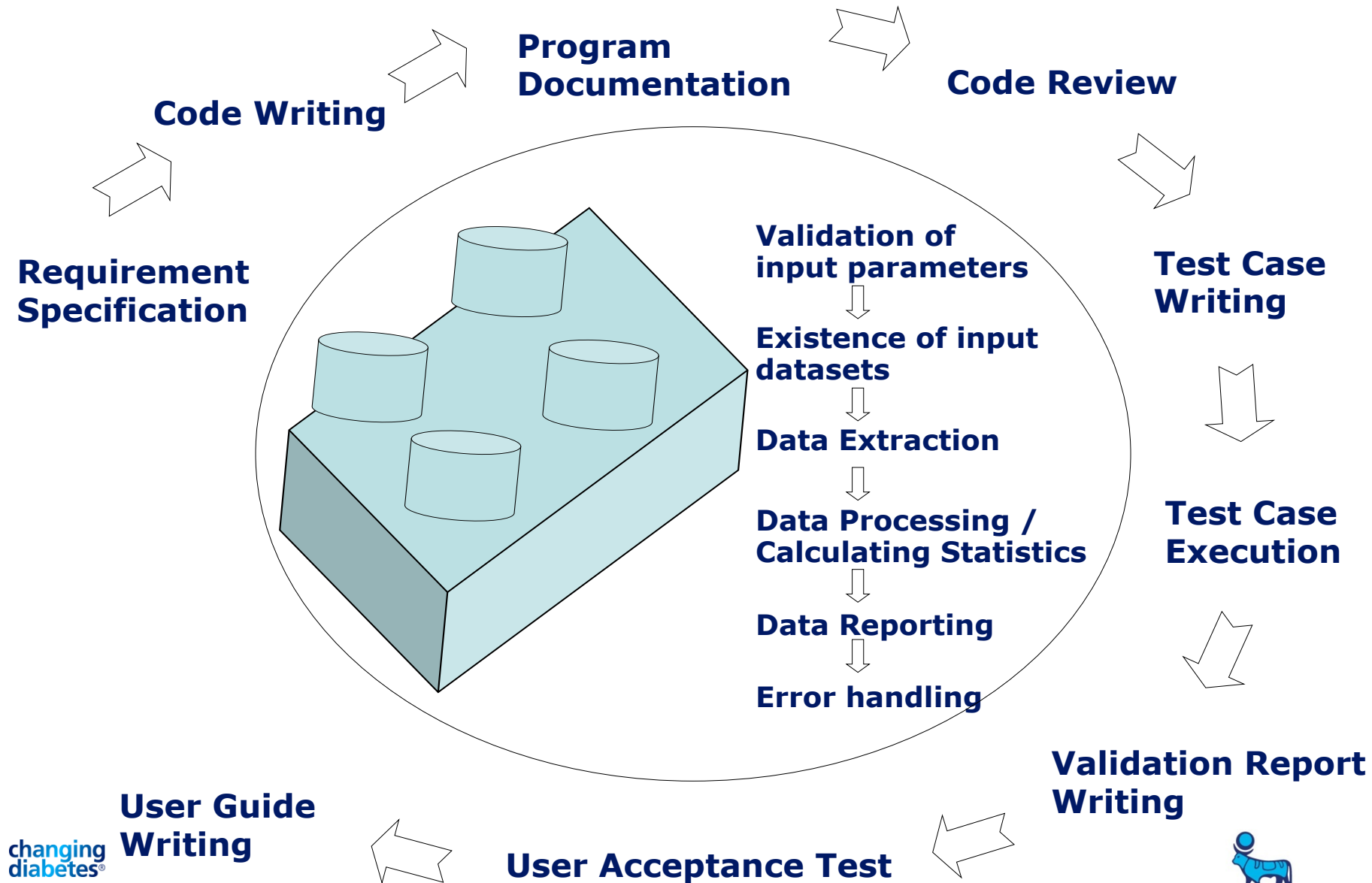
- 
- More than 30,000 employees in 74 countries
 - A world leader in diabetes care since 1923
 - Also leading positions in:
 - Haemostasis management
 - Growth hormone therapy
 - Hormone replacement therapy
 - Total net turnover (2010): 11.14 billion USD
 - Committed to financial, environmental and social results



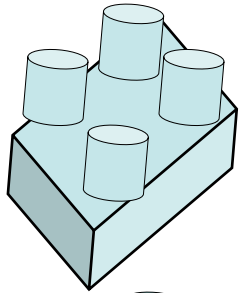
What is Building Block macro (BBM)?

- Building block macro (BBM) is basically like a **LEGO® brick** independently developed following the standard implementation methodology and a module based approach making sure it fits in the bigger picture of developing a Standard Program library ensuring **consistency, portability, reusability** and **easy maintenance**.
- For analysis and reporting in the Clinical Data Warehouse system, SAS Building Block macro components are developed and validated to be used in Novo Nordisk trials across R&D centres, indications and therapeutic areas.
- The idea behind this framework is to design sub-components that can be **reused** in trial-specific programs and across SAS standard programs in order to **minimize overall effort** in programming, documenting, validating and maintaining the standard program library.

Building block macro principle or the “Lego®” brick principle.

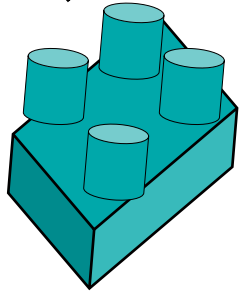


Categories of Building Block macro (BBM)



Initialization BBM:

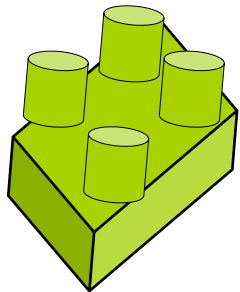
`%access_v01.sas`: Set up autocall libraries.



Data Extraction BBM:

`%extract_find_v01.sas`: Extraction of finding information.

`%extract_event_v01.sas`: Extraction of event information.

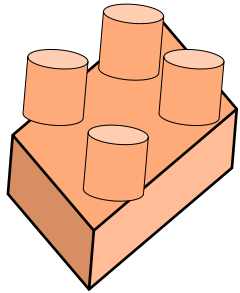


Data Processing BBM:

`%calculate_catgo_stats_v01.sas`: Frequency Statistics.

`%calculate_num_stats_v01.sas`: Calculation of statistics and percentiles.

Categories of Building Block macro (BBM)

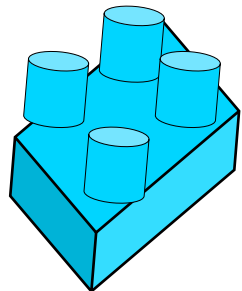


Data Reporting BBM:

`%print_summary_v01.sas`: Printing summary using PROC REPORT.

`%print_listing_v01.sas`: Printing listing using PROC REPORT.

`%print_mean_plot_v01.sas`: Printing mean plot using PROC GPLOT.



Derived Variables and Enrichments (DVE) BBM:

`%mk_age_groups_v01.sas`: Grouping subject by age.

`%mk_dve_partial_date_v02`: handling of partial dates.

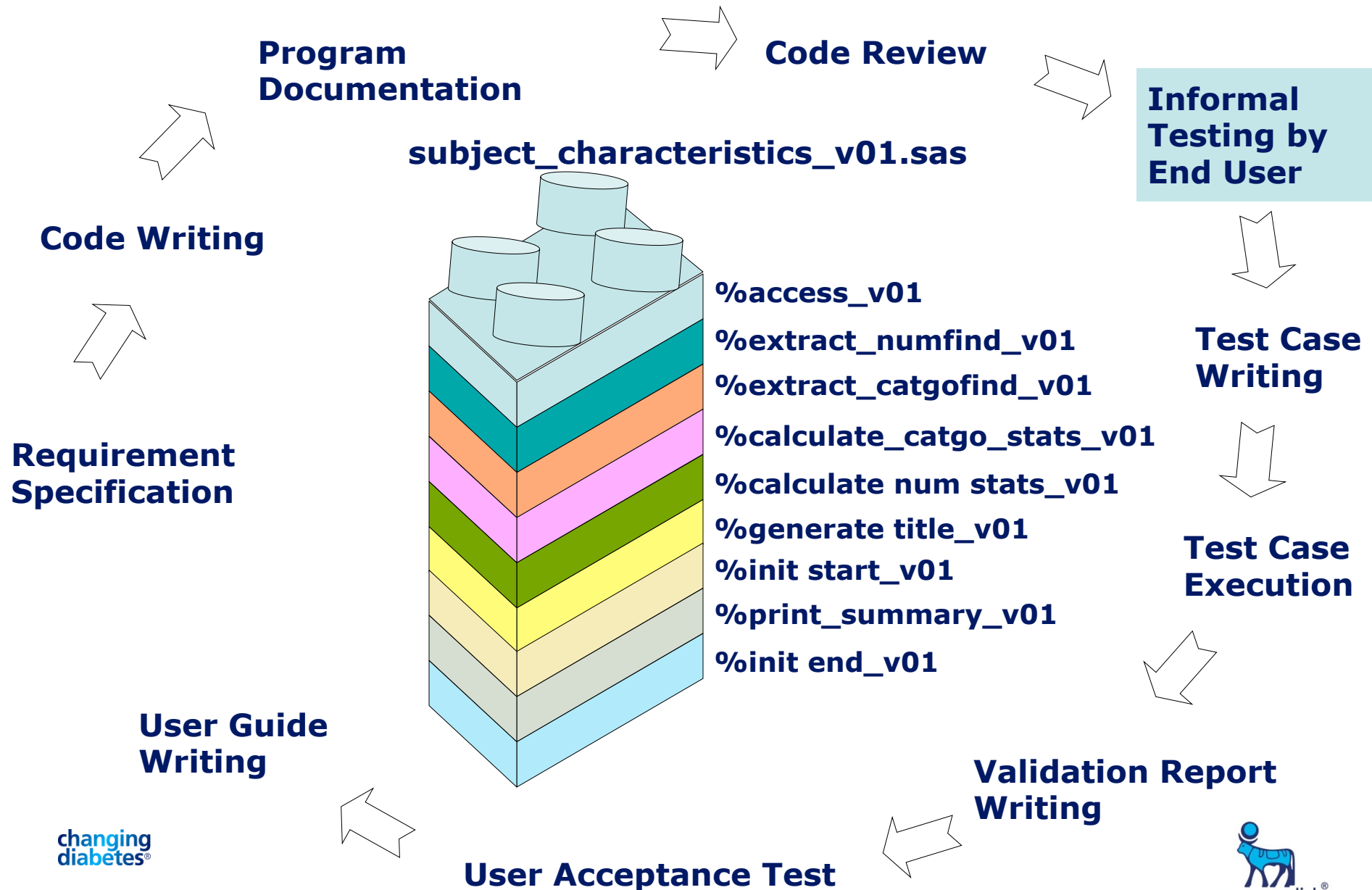
`%mk_dve_trt_emergent_v01`: Setting treatment emergent flag.

`%mk_dve_hypo_count_v01`: Count Hypoglycaemic Classification

Application of Building Block Macro Principle

Approach 1

Developing SAR (Statistical Analysis and Reporting) programs



Benefits:

- Consistent Output across project.
- Reusability.
- No Program Documentation needed.
- Time is saved from routine programming.
- Opportunity to develop standard utilities.
- Time is saved from routine programming.

Challenges:

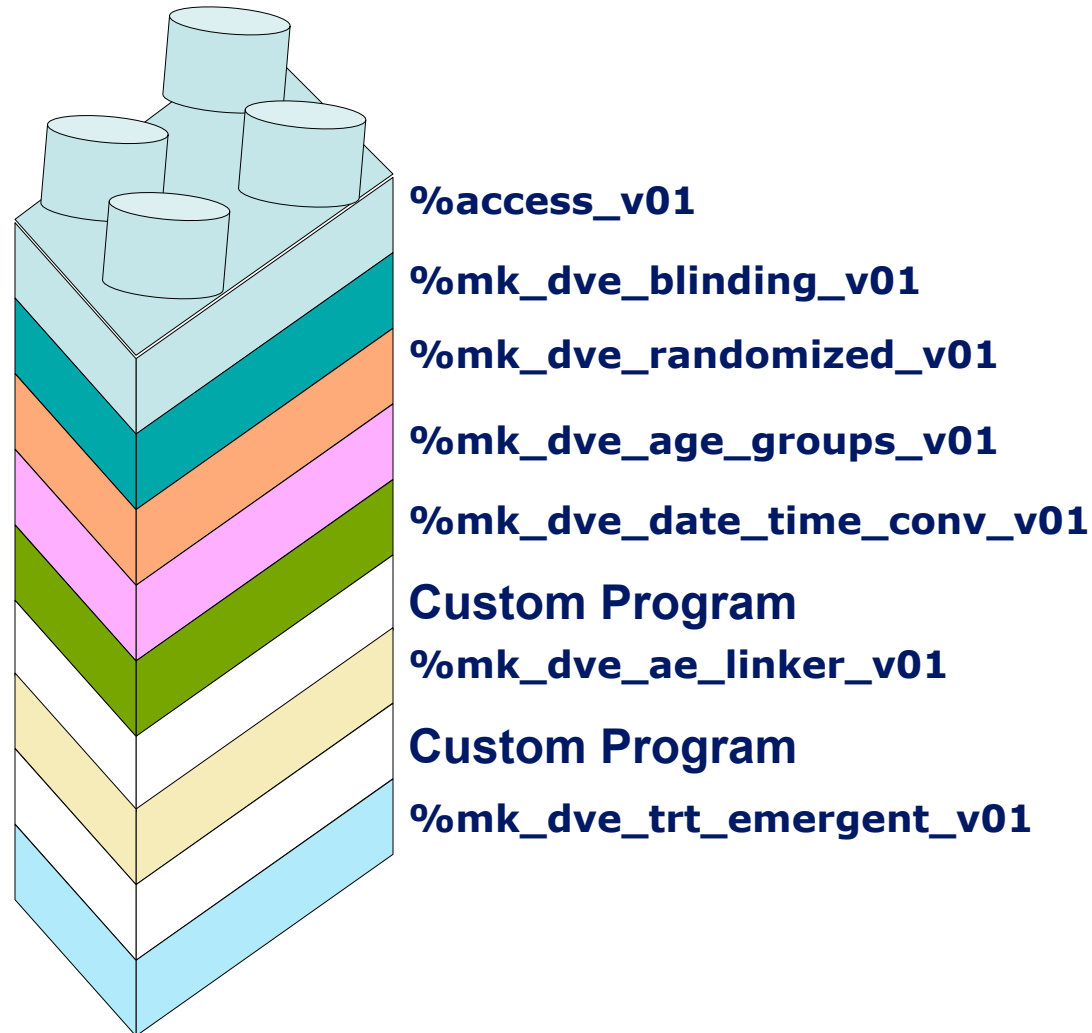
- Lack of Control
- Custom changes not possible.
- Debugging is a challenge when an error occurs.
- Dependence on the Standard Program Developer team for changes.
- Quick changes are not feasible.

Application of Building Block Macro Principle

Approach 2

Developing DVE (Derived Variables and Enrichments) programs

Custom Framework for DVE



Benefits:

- More control to the trial programmer.
- Users can be creative.
- Less Dependency on Standard Program Developer team.
- Quick fixes are possible in the White Box area.

Challenges:

- Complexity is high.
- Time Consuming.
- Experience is essential.
- More Documentation and Review work.

End User Survey:

It was seen in the survey that

- Experienced users are **happy** with DVE , less with SAR.
- Main advantage in using standard components is seen as **reusability** and **consistency**
- Main disadvantage in using standard components is seen as **lack of control** and **time consuming**
- Only few **SAR building blocks** are used in **custom** SAR programming.
- Lack of **Trust** was seen while using the SAR components.

What we do to improve the way that we work with standards

- Training, Training, Training and also Training!
- Improve toolbox of reference help and user guides.
- Increase Knowledge Sharing through active user forum.
- Feedback and survey from the end user.
- Listening to our end user.
- Job swap.

Challenges As a Standard Program Developer

- ✓ Development of standard programs is a difficult and **time-consuming exercise**.
- ✓ **Versioning** and maintaining of Standard Program and Building Block macro.
- ✓ There is **more paper work** involved in standards management, e.g. bugs reported in IT system management (Remedy) and change request raised for statistical standards.
- ✓ It can also be difficult to have the necessary **resources** as clinical projects often have the priority.

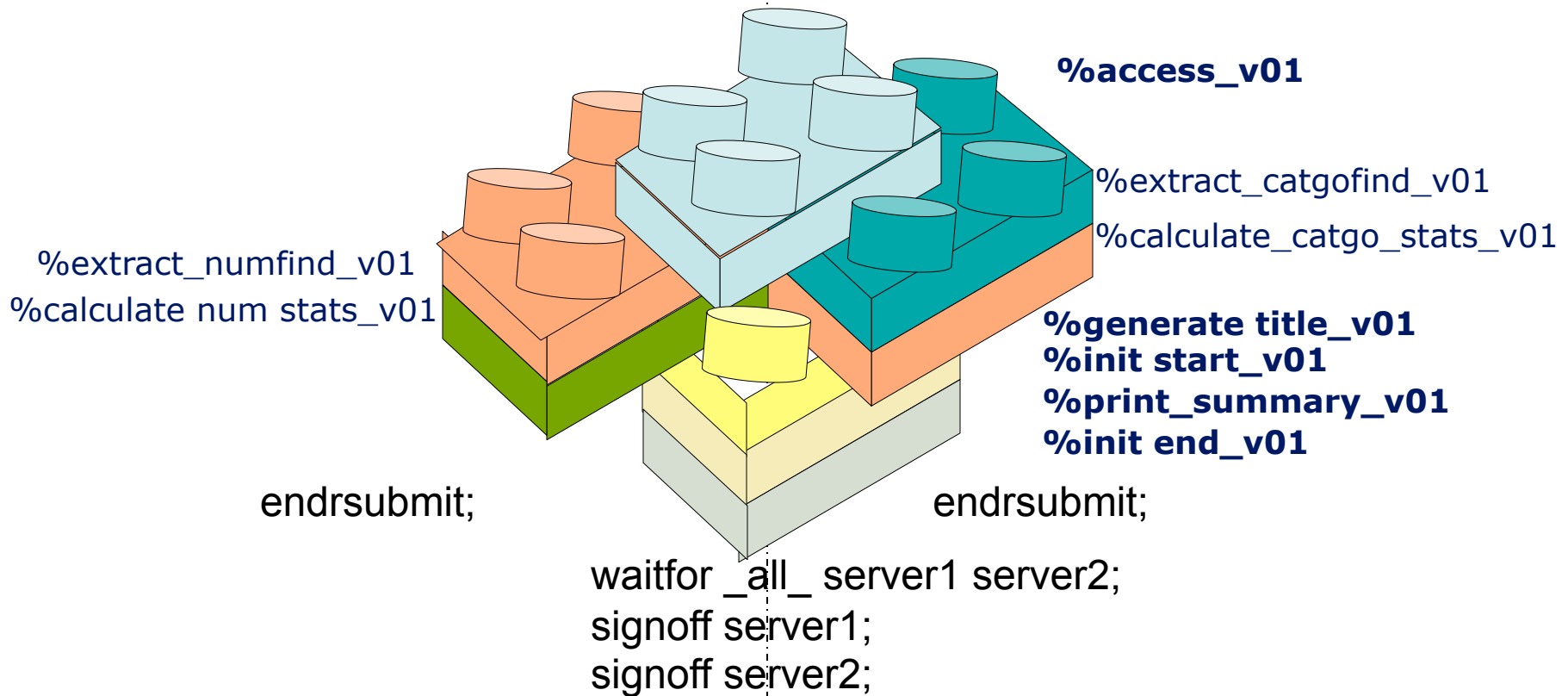
The future approach

Future Approach

subject_characteristics_v01.sas

```
signon server1 sascmd=...;
rsubmit server1 wait=no;
```

```
signon server2 sascmd=..;
rsubmit server2 wait=no;
```



To use the Power of SAS parallel processing in order to best utilize the UNIX box with 32 processor.



THANK YOU

For further questions, please write me at
frd@novonordisk.com