

# Dynamic Configurations

## Dynamic Configurations

Lyncode is developing a new implementation for the ConfigurationManager. Features:

- Dynamic configurations (runtime changes)
- Improved API (omission handling)
- Derived values support
- Modules support
- File Based

**NOTE:** Our team have internally discussed about a database based implementation (<https://wiki.duraspace.org/display/DSPACE/Database+Persistence+of+Configuration+State>), but it would have conceptual problems. Mainly:

- How to get the database configuration? (it must be placed somewhere - not in the database)
- Reading configurations from there will put an avoidable extra task on top of the DB
- On database crashes, some services (that, today, do not use database at all), will also crash
- Derived values aren't advisable (it would make many SQL queries to get a simple configuration value)

### API Changes : New available methods

```
String ConfigurationManager.getString(String key, String default);
String ConfigurationManager.getString(String module, String key, String default);
List<String> ConfigurationManager.getList(String key);
List<String> ConfigurationManager.getList(String module, String key);
```

### Core Changes : New possible features

It's now possible to easily set/add properties and save the respective files, also it's easy to access and modify the properties description.

### Pull Request

<https://github.com/DSpace/DSpace/pull/61>

**NOTE:** The under development SpringUI and its associated features made us recognize that, changing the core ConfigurationManager, is an important change to make DSpace more user friendly in the future.