

Toggle menu
Blue Gold Program Wiki

Navigation

- [Main page](#)
- [Recent changes](#)
- [Random page](#)
- [Help about MediaWiki](#)

Tools

- [What links here](#)
- [Related changes](#)
- [Special pages](#)
- [Permanent link](#)
- [Page information](#)

Personal tools

- [Log in](#)

personal-extra

Toggle search

Search

Random page

Views

- [View](#)
- [View source](#)
- [History](#)
- [PDF Export](#)

Actions

Module:Namespace detect/data

From Blue Gold Program Wiki

< [Module:Namespace detect](#)

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

Documentation for this module may be created at [Module:Namespace detect/data/doc](#)

```

-----
---
--
-- Namespace detect data
--
-- This module holds data for [[Module:Namespace detect]] to be loaded per
--
-- page, rather than per #invoke, for performance reasons.
--
-----
---

local cfg = require('Module:Namespace detect/config')

local function addKey(t, key, defaultKey)
    if key ~= defaultKey then
        t[#t + 1] = key
    end
end

end

-- Get a table of parameters to query for each default parameter name.
-- This allows wikis to customise parameter names in the cfg table while
-- ensuring that default parameter names will always work. The cfg table
-- values can be added as a string, or as an array of strings.

local defaultKeys = {
    'main',
    'talk',
    'other',
    'subjectns',
    'demospace',
    'demopage'
}

local argKeys = {}
for i, defaultKey in ipairs(defaultKeys) do
    argKeys[defaultKey] = {defaultKey}
end

for defaultKey, t in pairs(argKeys) do
    local cfgValue = cfg[defaultKey]
    local cfgValueType = type(cfgValue)
    if cfgValueType == 'string' then
        addKey(t, cfgValue, defaultKey)
    elseif cfgValueType == 'table' then
        for i, key in ipairs(cfgValue) do
            addKey(t, key, defaultKey)
        end
    end
    cfg[defaultKey] = nil -- Free the cfg value as we don't need it any
more.
end

```

```

local function getParamMappings()
    --[[
    -- Returns a table of how parameter names map to namespace names. The
keys
    -- are the actual namespace names, in lower case, and the values are
the
    -- possible parameter names for that namespace, also in lower case.
The
    -- table entries are structured like this:
    -- {
    --   [''] = {'main'},
    --   ['wikipedia'] = {'wikipedia', 'project', 'wp'},
    --   ...
    -- }
    --]]
    local mappings = {}
    local mainNsName = mw.site.subjectNamespaces[0].name
    mainNsName = mw.ustring.lower(mainNsName)
    mappings[mainNsName] = mw.clone(argKeys.main)
    mappings['talk'] = mw.clone(argKeys.talk)
    for nsid, ns in pairs(mw.site.subjectNamespaces) do
        if nsid ~= 0 then -- Exclude main namespace.
            local nsname = mw.ustring.lower(ns.name)
            local canonicalName =
mw.ustring.lower(ns.canonicalName)
            mappings[nsname] = {nsname}
            if canonicalName ~= nsname then
                table.insert(mappings[nsname], canonicalName)
            end
            for _, alias in ipairs(ns.aliases) do
                table.insert(mappings[nsname],
mw.ustring.lower(alias))
            end
        end
    end
    return mappings
end

return {
    argKeys = argKeys,
    cfg = cfg,
    mappings = getParamMappings()
}

```

Retrieved from

["https://www.bluegoldwiki.com/index.php?title=Module:Namespace_detect/data&oldid=220"](https://www.bluegoldwiki.com/index.php?title=Module:Namespace_detect/data&oldid=220)

Namespaces

- [Module](#)
- [Discussion](#)

Variants

This page was last edited on 18 February 2020, at 14:18.

Blue Gold Program Wiki

The wiki version of the Lessons Learnt Report of the Blue Gold program, documents the experiences of a technical assistance (TA) team working in a development project implemented by the Bangladesh Water Development Board (BWDB) and the Department of Agricultural Extension (DAE) over an eight+ year period from March 2013 to December 2021. The wiki lessons learnt report (LLR) is intended to complement the BWDB and DAE project completion reports (PCRs), with the aim of recording lessons learnt for use in the design and implementation of future interventions in the coastal zone.

- [Privacy policy](#)
- [About Blue Gold Program Wiki](#)
- [Disclaimers](#)

Developed and maintained by Big Blue Communications for Blue Gold Program



[Blue Gold Program Wiki](#)