Helo Blod's quick and easy toolkit on bilingual software for developers

This simple toolkit will help your IT project offer a great service in both Welsh and English.

It's for anyone or any organisation who wants to create a great experience for users in both Welsh and English.

You may be going through a procurement process for a service that needs to be bilingual – this document will help you with that too.

Remember: multilingual support is already common internationally, so using this document should be easy. Bear in mind too that you should include support for both English and Welsh (what we call "Bilingual capability" below) from the outset among other requirements being considered, such as accessibility, security, and legislation.

We've written this document with the support of Interceptor Solutions Ltd. We'd welcome any information or constructive feedback that may help us improve it over time. If you feel that the document could be clearer, or you think the system you're dealing with can't offer bilingual capability, have a look at the Welsh Language Commissioner's detailed guidance. And if you can't find the answer there, let heloblod@gov.wales know and we'll give you a helping hand.

Remember, it's almost always possible to find a way to do something to help bilingual capability, and if it's not, we may be able to work with the provider to find the solution! Remember: Helo Blod's got a long list of Welsh language/bilingual IT resources that you can use and reuse—from speech technology to Welsh maps, from translation to interfaces and lots more. Pop over to www.gov.wales/heloblod to download them for free.

We want to double the daily use of Welsh by 2050. That's a big ambition, so we've got big plans. Technology's a huge part of this and this guidance will help us all get there.

Glossary of Terms

Term	Definition
Application	A 'software application' (application or app for short) is computer software designed to perform a group of coordinated functions, tasks, or activities for the benefit of the user. In this document, we talk about 'applications' in a broad sense - that includes websites. These include all that have a natural language interface or that manage/process natural language information.
Bilingual	When we say 'bilingual' in this document, we mean the ability for a software application to support two natural languages in an equivalent manner. In Helo Blod's case, these are specifically Welsh and English; but you can apply the principles other multilingual scenarios too.
Detail	This means further information on how you'll be interpreting and applying the requirement. Consider this as part of the definition of the requirement.
Requirement	Together with the 'detail', the requirement defines the behaviour and capability of an application for it to conform to these guidelines.
Priority	• **M**(ust): Essential. If not met, the application cannot be considered bilingual • **S**(hould): If this requirement is not met, the application may be deficient in its bilingual capability. • **C**(ould): Where possible these requirements should be met to provide the best bilingual capability.
Language assets	The wording used in software interface labels, notifications, form fields, etc.
Natural language	The textual or voice representation of human languages used in or by applications.
User interface	How a user interacts with the application. This typically involves text or voice. When we say 'user interface' or 'UI' for short, in this document, we mean any form of user interface that involves the use of natural language.

1. Non-functional requirements

Definition

In this document, we talk about 'applications' in a broad sense - that includes websites. These include all that have a 'natural language' interface or that manage/process natural language information.

These requirements don't specify the functional behaviour of the application. i.e. they relate to the project, linguistic quality, implementation process, etc., and not any specific executive task the application may perform.

Are you managing a technology project that could benefit from this toolkit? Before starting any work, remember to check out and be sure of any legal requirements, e.g. the Welsh Language Standards regulations or a Welsh Language Scheme for. When going through each requirement, you'll need to make sure, in the context of the above, whether the requirement is a 'Must', a 'Should' or a 'Could'. If you're using this toolkit as part of a procurement process, make it easy for potential contractors by completing the priority column before publishing an invitation to tender.

KEY for priority column:
M = Must
S = Should
C = Could

Id	Requirement	Priority	Detail	Note how you'll achieve this
1.1	All requirements relating to bilingual capability and Welsh language support shall be handled with the same importance as other essential requirements.		This is to ensure that these capabilities are built into a project from the outset and are treated the same as any other requirements. The specific activities will vary depending on the nature of the project (i.e. assessing an existing system compared to the development of a new system), but will include: • Project Management • Requirement Management • Design and Development • Testing • Documentation • Acceptance	

Id	Requirement	Priority	Detail	Note how you'll achieve this
			For the assessment of an off-the-shelf solution, this may be met by simply using these requirements in a specification. Where the project involves the analysis and development of a new solution, this requirement needs more consideration.	
1.2	Both languages must be fully and equally available to a user before the application is used operationally.		This applies both to first use and to any upgrades. It prevents a situation where the application is available in one language with the other coming later, or a deficiency in equal provision of both languages.	
1.3	Linguistic quality should be of the same level for both languages. Standardised resources should be used and translation should only be provided by an accredited translator.		Where Welsh translations are provided, the translator must be a member of a recognised translation organisation. Where terminology and other language resources can be verified against a standardised resource e.g. databases of standardised terms and/or place names), this must be done. Terminology should be consistent across the whole application, including any associated user assistance content. You should construct or follow established style guide for localisation of software into Welsh. E.g. the Microsoft Language Portal Style Guide for Welsh and/or the Welsh Government's Arddulliadur (Style Guide) on the Byd Term Cymru website.	
1.4	Abbreviations and acronyms, e.g. in calendars, must be appropriate for each language.		Specifically: • The grammatical structure of the Welsh needs to be accounted for (e.g. with ordinals); • Digraphs (two characters forming one letter such as Ch, Ll, Ng, etc.) must be fully supported, i.e. both characters must be used in abbreviations.	
1.5	Dates need to be shown correctly in both languages.		Note: *locale* sensitivity can cause varying formats.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
1.6	Testing must ensure equality between languages and include language quality checks. This should be performed by qualified speakers of each language.		Testing (and formal test and acceptance cases where these are used) mustn't only consider the quality of the language being used, but also ensure that both languages are supported equally with the functionality and user experience being the same and of equal quality in either language. User testing should also include all kinds of end users, which includes, of course, Welsh speakers.	
1.7	Both languages need to be given equal consideration for design and presentation.		All branding, graphics and imagery should be either language-neutral or treat the Welsh language no less favourably than English. Where language is embedded within user Interface objects (e.g. images), this should either be language-neutral, in both languages or with a different (but equivalent) language-specific object for each language.	
1.8	The cost of any language provision should be factored into budgets and timeframes.		Consideration should be given not only to the cost of translation, licensing and other means to provision the language resources, but also to integration, testing and other project activities.	
1.9	User assistance content (i.e. help) should be available in Welsh and English. The Welsh version should be treated equally to English.		This requirement is specifically about the translation and availability of help and guidance content into each language. Where a user assistance subsystem includes functionality, such as navigation, searching, feedback, etc., this functionality will need to be fully bilingual in the same manner as all other aspects of the application. As for all other user interface content, user assistance content needs to be equally maintained in both languages when it's updated. Any resources embedded in this content, such as screen shots, need to be in the correct language.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
1.10	Any related applications (e.g. installers, support tools, etc.) with a user interface need to be identified and the guidelines in this document applied to them.		All applications that will be part of the intended solution must be compliant with these guidelines.	

2. Language provision and maintenance

Requirements relating to how language (i.e. translations/localisations) are provisioned and maintained for the application.

Id	Requirement	Priority	Detail	Note how you'll achieve this
2.1	Where a user can customise the language in the user interface, this capability must be available equally in both languages.		If a user of the application has the capability to maintain the user interface language (e.g. to customise labels, column headers, content, guidance, etc.), then this capability needs to be equally available for both languages.	
2.2	If regular/routine translation will be part of the operation of the application, functionality to support this process is highly desirable to avoid any delays. Receiving late translations may affect timetable/deadline.		This relates to any ongoing operational need to maintain user interface language resources. The effort in doing this can be mitigated by the ability to notify a translator of changes, automated translation workflow, provide a user interface specific to the purposes of language maintenance and other similar functionality.	
2.3	A well-defined, repeatable and, ideally, automated interface/process should exist to enable natural language resources to be identified, inventoried, extracted for translation (where relevant) and re-imported.		Consideration should be given to the effort (and hence cost) of extracting language assets from an application for translation and/or localisation, the management of the translation/localisation process and the re-import of translated/localised resources into the format required by the application. Where applicable, automated and repeatable processes will not only reduce cost but improve maintenance and help ensure higher quality and consistency. For textual content this will include direct integration with translation tools (translation workflow, translation memory, etc.) or a data interface that expedites the use of such tools. This is not a 'Must' requirement. But any cost of integrating and keeping translations up to date should be considered. It should	

Id	Requirement	Priority	Detail	Note how you'll achieve this
			also be noted that translation is a technical skill and should only be undertaken by competent and qualified human translators.	

If this guidance is not clear enough, or you think the system can't comply, drop us a line on heloblod@gov.wales to get practical advice. There is almost always a way, and if there isn't we may be able to work with the provider to find a way.

Version 3 – October 2023

3. Character set, encoding and *locale* considerations

There are differences between the Welsh and English alphabets: the set of letters, digraphs (two characters forming one letter), diacritics (accents/modifiers) and sort orders. These requirements ensure that both alphabets and character encodings are fully and equally supported.

Id	Requirement	Priority	Detail	Note how you'll achieve this
3.1	The full Welsh character set, including all diacritics, is to be supported in all aspects of the user interface, data processing and storage.		This relates to all textual outputs from the application that an end user can see. This includes the graphical User Interface, reports, emails and any other form of text. Unicode (UTF-8 or UTF-16) is recommended, but whatever encoding is used, it must support all Welsh characters with diacritics, i.e. accent, grave, circumflex and diaresis on all 7 vowels (a, e, i, o, u, w, y) in both upper and lower case: 56 characters in total. Good test cases are the ŵ and ŷ characters as these are absent in certain encodings.	
3.2	The character set encoding and the language used is to be explicitly defined wherever possible, including the user interface, documents, data interfaces and anywhere else textual data is used, stored or transmitted.		For text display, storage and definition formats it's usually possible to include an explicit descriptor of the encoding to be used, e.g. • Meta-data in HTML and XML for encoding and language (UTF-8 & xml:lang) and the equivalent for other technologies; • Operating platform configuration for Unicode/UTF-8; • Unicode/UTF-8 encoding to be used for all file formats; • In data interfaces to other applications and third parties.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
			This is essential to ensure that the full Welsh character set is supported despite any default setting on the application or user operating platform.	
3.3	Any *locale* usage should work equally for both languages.		If *locales* are used, they must include both English and Welsh, and comply with standards (ISO-639, ISO-3166, LCID, etc.).	
3.4	Diacritics should not affect sort order.		Diacritic marks (i.e. circumflex, acute, grave and diaresis) should not affect sort order. E.g. e, ê, é, è and ë are all equivalent when sorting.	
3.5	Alphabetically sorted lists must be shown in the correct order for the language.		A simple translation of an English user interface into Welsh will not work for an alphabetically sorted list. It's necessary to re-sort the translated text. E.g., a list of Welsh counties in English will place 'Anglesey' at the top of the list, whereas when translated into Welsh, the equivalent 'Ynys Môn' should appear at the bottom of the list.	
3.6	Sort Order		When an alphabetically sorted list is presented to a user it needs to use a sort order correct for the language being used. As lists can often use a mix of languages (e.g. proper nouns, etc.), the recommended approach for bilingual sorting is to use a 'superset' alphabet (i.e. a merge of the two alphabets) as outlined below in **bold**: **a, b, c, ch, d, dd, e, f, ff, g, ng, h, i, j, k, l, ll, m, n, o, p, ph, q, r, rh, s, t, th, u, v, w, x, y, z**	

4. Language selection and preference

Bilingual applications not only need to support both Welsh and English, they also need to allow immediate and easy switching between languages throughout their use.

They also need to enable a user to express their language preference and track it. It's essential that this capability is delivered in a manner that doesn't presume or oblige a preference for one language over the other.

Id	Requirement	Priority	Detail	Note how you'll achieve this
4.1	When the application is *first* accessed and the user language preference has not been explicitly selected by the user the application will: • Use a bilingual 'splash' selector, and/or; • Maintain the language from the referring content, and/or; • Implicitly determine the language from the method used to access the application; • Use the *locale* setting from the operating environment.		When a user first accesses the application, i.e. prior to an explicit language selection being made by the user, the language preference needs to be determined. It's essential that every attempt is made for the user intention to be identified (an implicit determination) or for the user to be prompted for their language preference as their first interaction (an explicit determination, e.g. a 'splash' selector). Often this can be done in the method by which the application is invoked, e.g. a link, domain name, shortcut, operating environment *locale*, user profile or menu option from a context where the language preference is already known. The key principle here is that both languages are equal (and therefore Welsh is not treated less favourable) and in no situation should the user interface be allowed to default to a one language or another. On implementation, this requirement will typically be refined to be specific to the nature of the application (i.e. web, voice, mobile, etc.) and the intended operating environment.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
4.2	An explicit language selection by a user will override any implicit or presumed selection.		Though an application will determine a user's language preference prior to an explicit selection by the user, once a user selects a language, this must override any implicit/presumed determination. E.g. operating system setting	
4.3	A language selector is to be immediately, equally and consistently available throughout the application. (and of same prominence for both languages).		Users should be able to change language at any time/stage of their interaction with an application. 'Immediately' is essential here. The language selector must be available without disrupting the engagement that a user has with the application, i.e. in a graphical interface it should be always be visible, in a voice interface an always available command, etc. The selector must be equally available for both languages, i.e. same location and prominence (font, size and colour) irrespective of which language is currently selected. Exception: A note must be documented of any exception e.g. when the user is using a small screen, it's possible due to the lack of space/number of characters to show the language selector as 'CY' or 'EN'.	
4.4	The language selector will be prominent and is accessible at any point/stage.		For a graphical ser interface this will be in the top right area of the interface. For other interface types it will need to be similarly prominent, consistent and easy to access.	
4.5	Changing language will maintain the user context.		Switching language should reload the same page. Some loss may occur, e.g. the data in a partially completed form or a partially completed voice dialogue, but otherwise, context should be maintained.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
			If any loss of context or state will inevitably occur beyond the loss of a partially completed form, the user should be notified of the impact and provided with an option not to change language.

The same is true for users the use a voice interface to complete forms. | |
| 4.6 | A language selection will be maintained throughout a user session. | | When a language selection, whether implicit or explicit, has been made that language will be used for the remainder of the user session until it's changed by an explicit user selection.

The user will be able to navigate and use the application entirely and solely in their selected language. | |
| 4.7 | A language selection should be stored and used as the default for future user sessions. | | When a language selection, whether implicit or explicit, has been made, that language should be stored in persistent storage (e.g. a cookie, device setting, other avenue etc.) so that it can be used as the default/initial language for a future session.

The stored language preference should be used as soon as the user accesses the application in the future as an explicit selection, therefore use of a user profile that is only accessed after a user 'logs in' (or otherwise authenticates or identifies themselves) is not an acceptable approach to meet this requirement. | |
| 4.8 | The language selector will be the word for the language *in that language* (i.e. "English" or "Cymraeg"). The Working Welsh/Iaith Gwaith logo (🟠) may be used alongside the | | As a second option, and only where space doesn't allow, a standard abbreviation may be used where space doesn't allow for the full word. I.e. EN or ENG for English and CY or CYM for Cymraeg. Should this option be chosen, provision should be made for the full name of the language to be accessed by screen readers and other accessibility software. | |

Id	Requirement	Priority	Detail	Note how you'll achieve this
	'Cymraeg' to represent a mechanism to switch content into Welsh (but not from Welsh to English).		National flags or other metaphors for language should not be used.	
4.9	Any language list will include the alternate language(s), i.e. it will not include the currently selected language.		When in the English User interface the language selector will be 'Cymraeg'. When in the Welsh user interface it will be 'English'. When in a multilingual context and there's a list of alternate languages, the Cymraeg option (when not in the Welsh user interface) will have the same prominence as English has (when not in the English Uuer interface).	
4.10	Application access methods must be bilingual and consistent with the selected language.		The means by which an application is accessed must be equivalent for both languages, i.e. either a language-neutral method or an equally available method in each language. For web-based applications, this is generally the domain name. Either the domain name should be language-neutral (e.g. a brand name) or there should be a domain name for each language. If there's a domain name for each language it can be used as the implicit indicator of language in the absence of an explicit user selection. When the user selects an alternate language the domain name used (or other relevant method for applications that aren't web-based) should be changed to one relevant to the selected language.	

5. Language equivalence, context and coverage

Equal treatment of both languages is fundamental. These requirements ensure that regardless of the language selected by the user there's no different in quality, coverage, resources or capabilities.

Id	Requirement	Priority	Detail	Note how you'll achieve this
5.1	All natural language in the user interface of the application will be exclusively in the selected language.		This addresses coverage. All user interface items must be exclusively in the selected language unless there's an overriding requirement. This means that all text must be in the selected language and no text in the alternate language should be visible, i.e. no 'mixed language' text. An exception to this is where an 'overriding requirement' explicitly needs mixed language text, e.g. in a language learning environment, translation functions, an application where mixed text is a specific and intended requirement, etc. When a language selection changes the language, all items should immediately change to the selected language. This includes text, voice, images and any other user interface assets that have a natural language component, whether an embedded part of the application or linked/integrated. It includes all aspects of the user interface, including those that are conditional or rarely used, such as error messages and notifications.	
5.2	Any integrated language support capabilities must have equivalent resources in each language.		This applies, for example, to spell checkers, thesauri, screen readers, voice recognition, voice and any other natural language resources, tools and assistance. This list is not exhaustive.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
5.3	User interfaces will be of identical structure, quality and functionality in each language with equivalent scope and quality of content.		Users will be able to fully navigate and use all application functionality in their selected language. There should be no loss of quality, scope of content, information or capability based on the language selection. E.g. messages/text from a member of the public should not be hidden in a forum when the interface language is different.	
5.4	The user interface arrangement must allow for variance of language length.		The quality of presentation must be the same in either language. Truncation and loss of content mustn't occur.	
5.5	Accessibility support and other content for use by third parties should be equivalent in both languages.		This is largely directed at screen readers for visually-impaired users. Not only should support be equal for both languages, but consideration should be given to the bilingual content. This extends to any consideration for access to the application and its content by external resources and includes content for semantic web purposes, search engine optimisation, device-specific formats (e.g. responsive websites), etc.	
5.6	Links and navigable references to external resources must lead to the resource in the same language as that currently selected by the user.		When an external resource is available in the currently selected user interface language, a link/reference to it must be to the version in the current language. Where the provision of the resource is within the scope of the application, or the organisation managing the application, the resource must be available in both languages.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
			Examples include (but aren't limited to) websites, social media, documents and other assets, help files, etc.	

If this guidance is not clear enough, or you think the system can't comply, drop us a line on heloblod@gov.wales to get practical advice. There is almost always a way, and if there isn't we may be able to work with the provider to find a way.

Version 3 – October 2023

6. Language data management

There is more to applications than just user interface and content. Applications typically receive process and publish data. Where this data contains language-specific content, the application should provide capabilities to manage this bilingually.

Id	Requirement	Priority	Detail	Note how you'll achieve this
6.1	Where the facility exists for a user to enter data in both languages, display ordering (i.e. precedence) and which fields are mandatory must be consistent with the currently selected language.		When enabling a user to enter information in both languages it may be the case that two sets of data entry fields are used, one for each language. This requirement addresses the prominence (ordering) and validation rules to be applied when bilingual variants of a data field are used. - Mandatory requirements for fields in each language must be equivalent, or based on the current selected user interface language; - Data validation must be the same, or based on the current selected user interface language; - Any prominence to one (language) set of fields will reflect the current selected user interface language. Note: eliminating unstructured data entry significantly reduces this requirement.	
6.2	When data that contains language-sensitive data is entered into/associated with an application by a method that allows the language to be determined, it must be possible to identify the language of that data.		This means that when accepting data entry from any interface (data or user) and where that interface allows for the language of the data to be identified (a field specific to the language, metadata in a data interface, etc.), then the application should tag that data with the relevant language code. This can be via metadata, other explicit data indicators, or it could be a matter of implicit language identification such as	

Id	Requirement	Priority	Detail	Note how you'll achieve this
			storing the data in a location (file system location, database field, etc.) that is specific to the language. This requirement applies to all forms of data, including data objects, files, etc.	
6.3	Search functions should work equally in either language.		Where an application provides a search facility it should work in either language, returning all relevant results for the terms entered, i.e. it should search across both Welsh and English content. Where content is absent in for the currently selected user interface language but is present in the alternate language then it should be displayed—in the absent of content, even if it's not available in the selected language is to be avoided at all costs to avoid disparity of functionality.	
6.4	Metadata and other structure data attributes should be available in both languages, even if some content for one language is absent.		I.e. metadata and other structured data attributes are to be associated across both languages so that they can be found by search even if some data attributes (e.g. free text, etc.) aren't available.	
6.5	User entry of unstructured natural language data should be limited to essential cases.		When a user enters natural language data in an unstructured manner (e.g. in a free text field), it may become difficult to manage this in a bilingual and cross-lingual manner. The process of managing, using and (potentially) publishing/displaying data is significantly simplified if it's acquired in a structured/tokenised manner, e.g. list and/or menu selection, options, user selected categories, yes/no responses, etc.	

Id	Requirement	Priority	Detail	Note how you'll achieve this
6.6	When data that contains language-sensitive data is entered into/associated with an application separately in each language, the application should link both items.		Data items should be considered as two representations of the same conceptual data item, expressed by a 'variation' in each of the languages. The link can be explicit, by use of metadata or implicit, e.g. storage location. This requirement applies to all forms of data, including data objects, files, etc.	
6.7	When data is provided by a third party, either via the user Interface or a data interface, it must be possible to require that it's to be provided bilingually.		This requirement may vary depending on the nature of the application and the nature of your requirements and so may be modified. Where data is provided by a third party that has a statutory obligation to operate bilingually, it's appropriate to require that data to be bilingual and to handle it accordingly. Where data is provided by a third-party organisation without a statutory obligation to operate bilingually, the data may or may not be provided bilingually and may be in either language. The validation rules and data handling will therefore vary with circumstance, but it's essential to ensure that the specified application will provide the required flexibility.	

7. Outputs and language considerations

Applications often produce outputs (emails, reports, documents, mail-merge letters, etc.) that have a lifespan beyond the period over which a user interacts with the application. If the user choses one language for the interface language, the application needs to be able to create documents in more than one language e.g. if the user choses Welsh as the interface for the MS Word, the user will be able to create English and Welsh documents.

Id	Requirement	Priority	Detail	Note how you'll achieve this
7.1	When recording information about an entity (e.g. an individual) which has a language preference, there must be the ability to record that information.		This should record the preference for each type of communication, e.g. voice, email, SMS, letter, etc.	
7.2	Standard outputs from an application, i.e. reports, need to be available bilingually.		This addresses both the standard language in the report, i.e. title, column headings, page indicators, etc. and also the data. Standard application text must be available bilingually. Where data is only available in one language, i.e. unstructured data entered monolingually by a user, this requirement doesn't apply to that data.	
7.3	Outputs provided for an entity (e.g. an individual or organisation) need to recognise and be tailored to the recorded language preference of that entity.		The specific approach taken will depend on the nature of the output. The application must both be sufficiently flexible to support all required approaches and be able to support bilingual outputs (with preferred language most prominent). Where data fields are embedded in the output (i.e. a mail-merge type of function), the appropriate language for each data field should be used.	
7.4	Communications such as emails produced by the		Specific capabilities that must be provided to meet this Requirement are:	

Id	Requirement	Priority	Detail	Note how you'll achieve this
	application must be bilingual.		• The application must support either/both of monolingual emails in each language or bilingual emails; • Subject lines and content for bilingual emails to have prominence/order of language based on language preference; • Reply addresses (address and domain) based on language preference; • Subject line modifiers (e.g. RE:/ATB:, YML/FW:, etc.) should be correct for the prominent language in the email; • Appended text (signatures, disclaimers, etc.) need to be bilingual or consistent with the language of the email; • Attachments must be consistent with the language and in the prominence order of the preferred language.	
7.5	Bilingual content is to include guidance for accessibility tools.		When content is provided in an electronic output (document, email, etc.) bilingually, it's essential that guidance for accessibility tools (such as screen readers) is embedded, e.g. 'skip to English text below', etc.	

8. Geographic information—language considerations

Where maps, addresses and other geographic related information is used by an application, it needs to account for the variants (note, these aren't generally direct translations) of place names in each language.

Id	Requirement	Priority	Detail	Note how you'll achieve this
8.1	Data overlaid onto maps that is language-sensitive must be available in each language with the correct language used for the selected language.		Applications will often overlay maps with symbols, images, text and other resources that have linguistic content. This information must be in the same language as the rest of the interface. This also applies to legends, directions and any other resources related to the map (though this is also covered by other requirements).	
8.2	Any maps used must have Bilingual place names to the extent possible.		This requirement is constrained to the availability of the digital maps with bilingual and/or Welsh-only place names. Whenever these are available they should be used.	
8.3	Use of Welsh Postcode Address Files		Postcode Address Files (PAFs), GeoPlace and other address verification solutions exist for both the English and Welsh variations of addresses in Wales. When the English Address Solution is used to lookup, validate or other address processing, the Welsh Address Solution also needs to be used to ensure that the Welsh variant of an address is used in the Welsh user interface, for Welsh language communications and for the validation of Welsh addresses.	