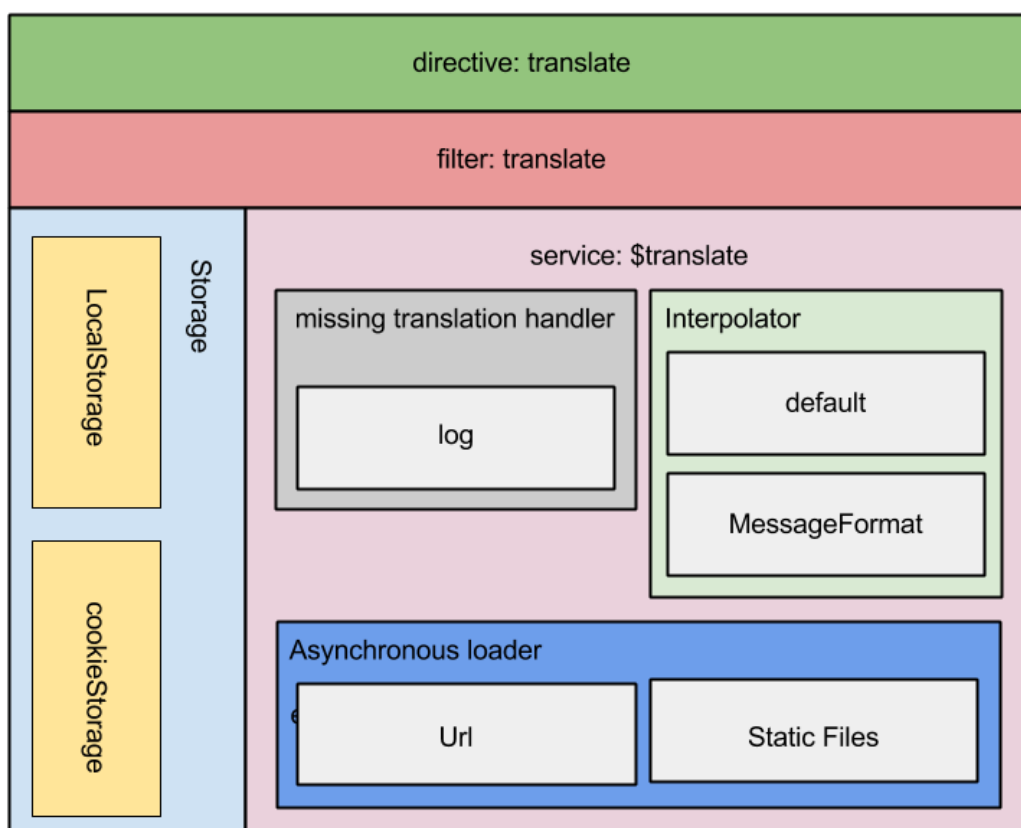


در [بخش قبلی](#) به معرفی ماژول angular-translate پرداختیم. در این بخش قصد داریم تا به بررسی مفهومی قابلیت های موجود در این ماژول بپردازیم.

شکل زیر یک شمای کلی را از قابلیت های angular-translate، نمایش می دهد. همانطور که ملاحظه می کنید در مواردی نظیر ذخیره سازی زبان ها، بارگذاری ریسورس های زبان ها و گرامر های استفاده در DOM راهکار های گوناگونی دارد.

angular-translate - Conceptual Overview



angular-translate دایرکتیو و فیلتر هایی را به صورت کامپوننت عرضه کرده است که شما می توانید به وسیله ی آنها پروژه خود را به زبان های گوناگون localize کنید. آنچه که در تصویر فوق مشاهده می کنید در حقیقت ساختار ماژول angular-translate را نمایش می دهد. به این صورت که در اولین لایه دایرکتیو translate قرار گرفته است.

سطح بعدی شاید کمی جذاب تر به نظر برسد. هر دو بخش دایرکتیو و فیلتر از سرویس تزریق شده \$translate استفاده می کنند. این بدین معنی است که در هنگام تغییر یک زبان به زبان دیگر به وسیله دایرکتیوهایی که در view نوشته شده است، در حقیقت شما این دایرکتیوها را به سطح فیلتر انتقال داده اید و پس از آن، فیلتر آن را به لایه سرویس برده و سرویس کار اصلی را انجام می دهد.

در سطح بعدی لایه Interpolator قرار گرفته است که وظیفه تغییر DOMها را بر عهده دارد. این فرآیند متغیرهایی را که در ریسورس قرار داده ایم، درایه به درایه در view و در مکان های هم نام قرار می دهد. این فرآیند به وسیله راهکارهای گوناگونی از قبیل Message Format و یا Angular Core Interpolation Service امکان پذیر است.

جعبه خاکستری رنگ بخش missing translation handler را نشان می دهد. این handlerها زمانی فراخوانی می گردند که translator به دنبال یک key می گردد که تعریف نشده باشد. angular-translate خود شامل یک logging service می باشد که به صورت extension باید آن را به پروژه خود اضافه نمایید که در بخش های بعدی در مورد آن مفصل تر بحث می کنیم.

بخش asynchronous loader شما را قادر می سازد که داده های زبان های متفاوت را به صورت غیر هم زمان بارگذاری نمایید. angular-translate از ماژول های asynchronous loader پشتیبانی می کند. روش معرفی شده در سایت مرجع در مورد بارگذاری غیر همزمان urlLoader و staticFilesLoader هستند که در بخش های بعدی به آن خواهیم پرداخت.

angular-translate برای ذخیره سازی داده های بازگذاری شده و نگه داشتن زبان سایت در صورت reload شدن صفحه راهکارهای متفاوتی را ارائه نموده است. این قابلیت در ابتدا چک می کند که ریسورس های زبان درون یک local storage قرار گرفته اند یا خیر. این عملیات طی یک فرآیند جستجوی key-value صورت می گیرد و در نهایت فایل های ریسورس زبان مورد نظر که درون کوکی یا localStorage ذخیره شده اند بارگذاری می شوند. angular-translate به صورت توکار دارای دو راهکار localStorage و cookieStorage برای ذخیره سازی ریسورس ها می باشد.

در بخش بعدی در مورد نحوه ی ذخیره سازی داده ها به دو روش cookieStorage و localStorage بیشتر صحبت خواهیم کرد.